Package ‘civis’

June 22, 2020

Title R Client for the 'Civis Platform API'

Version 3.0.0


Depends R (>= 3.2.0)

License BSD_3_clause + file LICENSE

LazyData true

URL https://github.com/civisanalytics/civis-r

BugReports https://github.com/civisanalytics/civis-r/issues

Imports future (>= 1.8.0), httr, jsonlite, methods, memoise, stats, utils

Suggests feather, ggplot2, knitr, rmarkdown, roxygen2, mockery, R.utils, rstudioapi, testthat, yaml

RoxygenNote 7.1.0

VignetteBuilder knitr


Encoding UTF-8

NeedsCompilation no

Author Patrick Miller [cre, aut], Keith Ingersoll [aut], Bill Lattner [ctb], Anh Le [ctb], Michelangelo D'Agostino [ctb], Sam Weiss [ctb], Stephen Hoover [ctb], Danning Chen [ctb],
R topics documented:

admin_delete_announcements ........................................... 18
admin_get_announcements ........................................... 18
admin_list_announcements ........................................... 19
admin_list_organizations ........................................... 20
admin_patch_announcements ........................................... 21
admin_patch_themes .................................................... 22
admin_post_announcements ........................................... 23
admin_post_themes ....................................................... 23
aliases_delete ............................................................ 24
aliases_delete_shares_groups .......................................... 25
aliases_delete_shares_users ........................................... 25
aliases_get ............................................................... 26
aliases_get_object_type ............................................... 26
aliases_list .............................................................. 27
aliases_list_shares ...................................................... 28
aliases_patch ............................................................ 29
aliases_post .............................................................. 30
aliases_put ............................................................... 31
aliases_put_shares_groups ............................................. 32
aliases_put_shares_users .............................................. 33
announcements_list ..................................................... 34
apps_delete_instances_projects ....................................... 35
apps_delete_instances_shares_groups ................................ 35
apps_delete_instances_shares_users ................................ 36
apps_delete_releases_shares_groups ................................ 36
apps_delete_releases_shares_users ................................. 37
apps_get ................................................................. 37
apps_get_instances ..................................................... 38
apps_get_releases ....................................................... 39
apps_list ................................................................. 40
apps_list_instances ..................................................... 40
apps_list_instances_projects ......................................... 41
apps_list_instances_shares ............................................ 42
apps_list_releases ..................................................... 43
apps_list_releases_shares ............................................ 44
apps_patch_instances .................................................. 45
apps_patch_releases ................................................... 46
topics documented:

- `apps_post_instances` .............................................................. 47
- `apps_post_releases` ............................................................... 48
- `apps_put_features` ................................................................. 48
- `apps_put_instances_archive` .................................................... 49
- `apps_put_instances_projects` ................................................... 50
- `apps_put_instances_shares_groups` ........................................... 51
- `apps_put_instances_shares_users` ............................................. 52
- `apps_put_releases_archive` ...................................................... 53
- `apps_put_releases_shares_groups` ............................................. 54
- `apps_put_releases_shares_users` .............................................. 55
- `await` ........................................................................ 56
- `cancel` ....................................................................... 58
- `CivisFuture` ................................................................. 59
- `civis_file` ................................................................. 61
- `civis_file_manifest` .............................................................. 61
- `civis_ml` ................................................................. 62
- `CIVIS_ML_CLASSIFIERS` ......................................................... 68
- `civis_ml_extra_trees_classifier` ............................................... 68
- `civis_ml_extra_trees_regressor` ................................................ 73
- `civis_ml_gradient_boosting_classifier` ..................................... 77
- `civis_ml_gradient_boosting_regressor` ....................................... 81
- `civis_ml_random_forest_classifier` ........................................... 86
- `civis_ml_random_forest_regressor` ............................................ 90
- `CIVIS_ML_REGRESSORS` ......................................................... 94
- `civis_ml_sparse_linear_regressor` ........................................... 95
- `civis_ml_sparse_logistic` ....................................................... 98
- `civis_ml_sparse_ridge_regressor` ........................................... 102
- `civis_platform` .......................................................... 106
- `civis_script` ............................................................ 107
- `civis_table` ............................................................. 108
- `civis_to_multifile_csv` ......................................................... 108
- `clusters_delete_kubernetes_partitions` .................................... 110
- `clusters_get_kubernetes` ...................................................... 110
- `clusters_get_kubernetes_instance_configs` .............................. 111
- `clusters_get_kubernetes_partitions` ........................................ 113
- `clusters_list_kubernetes` ...................................................... 114
- `clusters_list_kubernetes_deployments` .................................... 115
- `clusters_list_kubernetes_deployment_stats` ............................ 116
- `clusters_list_kubernetes_instance_configs_historical_graphs` .... 117
- `clusters_list_kubernetes_instance_configs_user_statistics` ......... 118
- `clusters_list_kubernetes_partitions` ........................................ 119
- `clusters_patch_kubernetes` .................................................... 120
- `clusters_patch_kubernetes_partitions` ...................................... 121
- `clusters_post_kubernetes` ..................................................... 122
- `clusters_post_kubernetes_partitions` ....................................... 123
- `coef.civis_ml` ............................................................... 124
- `credentials_delete_shares_groups` ......................................... 124
- `credentials_delete_shares_users` .......................................... 125
- `credentials_delete_shares_users` .......................................... 125
topics documented:

- credentials_get .................................................. 126
- credentials_list .................................................. 126
- credentials_list_shares ......................................... 128
- credentials_post .................................................. 129
- credentials_post_authenticate .................................. 130
- credentials_post_temporary ..................................... 131
- credentials_put .................................................. 131
- credentials_put_shares_groups ................................. 132
- credentials_put_shares_users ................................. 133
- databases_delete_whitelist_ips ............................... 134
- databases_get .................................................. 135
- databases_get_whitelist_ips .................................... 135
- databases_list .................................................. 136
- databases_list_advanced_settings ............................. 137
- databases_list_schemas ....................................... 137
- databases_list_tables ......................................... 138
- databases_list_tables_search ................................. 139
- databases_list_whitelist_ips .................................. 139
- databases_patch_advanced_settings ......................... 140
- databases_post_schemas_scan ................................. 141
- databases_post_whitelist_ips ................................. 142
- databases_put_advanced_settings ............................. 143
- default_credential ............................................... 144
- download_civis .................................................. 144
- endpoints_list .................................................. 146
- enhancements_delete_cass_ncoa_projects .................... 146
- enhancements_delete_cass_ncoa_runs ........................ 147
- enhancements_delete_cass_ncoa_shares_groups .............. 147
- enhancements_delete_cass_ncoa_shares_users ............... 148
- enhancements_delete_cass_ncoa_runs_logs .................. 149
- enhancements_delete_cass_ncoa_runs_outputs ............... 150
- enhancements_delete_civis_data_match_projects ............ 149
- enhancements_delete_civis_data_match_runs ............... 149
- enhancements_delete_civis_data_match_shares_groups .... 150
- enhancements_delete_civis_data_match_shares_users ....... 150
- enhancements_delete_geocode_projects ...................... 151
- enhancements_delete_geocode_runs ......................... 151
- enhancements_delete_geocode_shares_groups ............... 152
- enhancements_delete_geocode_shares_users ............... 152
- enhancements_get_cass_ncoa ................................. 152
- enhancements_get_cass_ncoa_runs ......................... 154
- enhancements_get_civis_data_match ......................... 154
- enhancements_get_civis_data_match_runs ................. 155
- enhancements_get_geocode .................................. 158
- enhancements_get_geocode_runs .............................. 160
- enhancements_list ............................................. 161
- enhancements_list_cass_ncoa_projects ..................... 162
- enhancements_list_cass_ncoa_runs ......................... 163
- enhancements_list_cass_ncoa_runs_logs .................... 164
- enhancements_list_cass_ncoa_runs_outputs ............... 165
R topics documented:

- enhancements_list_cass_ncoa_shares .............................................. 166
- enhancements_list_civis_data_match_projects .................................... 167
- enhancements_list_civis_data_match_runs ........................................ 168
- enhancements_list_civis_data_match_runs_logs .................................. 169
- enhancements_list_civis_data_match_runs_outputs ............................... 170
- enhancements_list_civis_data_match_shares ..................................... 171
- enhancements_list_field_mapping .................................................. 172
- enhancements_list_geocode_projects .............................................. 172
- enhancements_list_geocode_runs ................................................... 173
- enhancements_list_geocode_runs_logs .......................................... 174
- enhancements_list_geocode_runs_outputs ....................................... 175
- enhancements_list_geocode_shares .............................................. 176
- enhancements_list_types .......................................................... 177
- enhancements_patch_cass_ncoa ..................................................... 177
- enhancements_patch_civis_data_match .......................................... 181
- enhancements_patch_geocode ...................................................... 185
- enhancements_post_cass_ncoa ...................................................... 188
- enhancements_post_cass_ncoa_cancel .......................................... 192
- enhancements_post_cass_ncoa_runs ............................................ 192
- enhancements_post_civis_data_match ........................................... 193
- enhancements_post_civis_data_match_cancel ................................... 197
- enhancements_post_civis_data_match_clone ................................... 197
- enhancements_post_civis_data_match_runs .................................... 200
- enhancements_post_geocode ....................................................... 200
- enhancements_post_geocode_cancel ............................................. 204
- enhancements_post_geocode_runs .............................................. 204
- enhancements_put_cass_ncoa ...................................................... 205
- enhancements_put_cass_ncoa_archive ......................................... 209
- enhancements_put_cass_ncoa_projects ......................................... 211
- enhancements_put_cass_ncoa_shares_groups .................................... 212
- enhancements_put_cass_ncoa_shares_users .................................... 213
- enhancements_put_civis_data_match ........................................... 214
- enhancements_put_civis_data_match_archive .................................. 218
- enhancements_put_civis_data_match_projects ................................ 220
- enhancements_put_civis_data_match_shares_groups ........................... 220
- enhancements_put_civis_data_match_shares_users ................................ 221
- enhancements_put_geocode ....................................................... 222
- enhancements_put_geocode_archive ............................................. 226
- enhancements_put_geocode_projects ........................................... 228
- enhancements_put_geocode_shares_groups .................................... 228
- enhancements_put_geocode_shares_users .................................... 229
- exports_get_files_csv ............................................................ 230
- exports_list ................................................................. 232
- exports_patch_files_csv .......................................................... 233
- exports_post_files_csv ............................................................ 235
- exports_put_files_csv .............................................................. 238
- exports_put_files_csv_archive ................................................... 240
- feature_flags_delete_groups ..................................................... 241
R topics documented:

feature_flags_delete_users ........................................ 242
feature_flags_get .................................................. 242
feature_flags_list .................................................. 243
feature_flags_put_groups .......................................... 244
feature_flags_put_users ........................................... 245
fetch_all ............................................................. 246
fetch_and_generate_client ......................................... 246
fetch_logs ............................................................. 247
fetch_oos_scores .................................................... 248
fetch_output_file_ids .............................................. 248
fetch_predictions .................................................... 249
fetch_until ........................................................... 250
files_delete_preprocess_csv ....................................... 251
files_delete_projects ............................................... 251
files_delete_shares_groups ........................................ 252
files_delete_shares_users .......................................... 252
files_get ............................................................ 253
files_get_preprocess_csv ......................................... 254
files_list_projects ............................................... 255
files_list_shares ................................................... 256
files_patch .......................................................... 257
files_patch_preprocess_csv ....................................... 258
files_post ........................................................... 259
files_post_multipart ................................................. 260
files_post_multipart_complete .................................... 261
files_post_preprocess_csv ........................................ 261
files_put ............................................................ 263
files_put_shares_groups .......................................... 264
files_put_shares_users ............................................. 265
files_put_projects ................................................. 266
files_put_shares_groups .......................................... 267
files_put_shares_users ............................................. 268
generate_client ...................................................... 269
get_database_id ....................................................... 269
get_error ............................................................. 270
get_feature_importance ............................................. 271
get_job_type_version ................................................. 271
get_metric ............................................................. 272
get_script_fun ......................................................... 272
get_status ............................................................. 273
get_table_id .......................................................... 274
get_template_ids_all_versions ................................... 274
get_train_template_id .............................................. 275
git_repos_delete .................................................... 275
git_repos_get ........................................................ 276
git_repos_list ....................................................... 276
git_repos_post ....................................................... 277
groups_delete ........................................................ 278
groups_delete_members ........................................ 278
groups_delete_shares_groups ................................ 279
groups_delete_shares_users .................................. 279
groups_get ...................................................... 280
groups_list .................................................... 281
groups_list_shares ............................................ 282
groups_patch ................................................... 283
groups_post ..................................................... 285
groups_put ...................................................... 287
groups_put_members ............................................ 289
groups_put_shares_groups .................................... 290
groups_put_shares_users ..................................... 291
hist.civis_ml .................................................. 292
imports_delete_files_csv ..................................... 293
imports_delete_files_runs .................................... 293
imports_delete_projects ..................................... 294
imports_delete_shares_groups ................................ 294
imports_delete_shares_users ................................ 295
imports_delete_syncs ........................................ 295
imports_get .................................................... 296
imports_get_batches ......................................... 298
imports_get_files_csv ....................................... 299
imports_get_files_runs ...................................... 300
imports_list ................................................... 301
imports_list_batches ........................................ 304
imports_list_files_runs ..................................... 305
imports_list_files_runs_logs ................................ 306
imports_list_projects ........................................ 306
imports_list_runs ............................................. 307
imports_list_runs_logs ....................................... 308
imports_list_shares ......................................... 309
imports_patch_files_csv .................................... 310
imports_post .................................................. 313
imports_post_batches ........................................ 317
imports_post_cancel ......................................... 318
imports_post_files .......................................... 319
imports_post_files_csv ..................................... 320
imports_post_files_runs .................................... 324
imports_post_runs ............................................. 324
imports_post_syncs .......................................... 325
imports_put .................................................... 329
imports_put_archive ......................................... 333
imports_put_files_csv ....................................... 335
imports_put_files_csv_archive ............................... 339
imports_put_projects ........................................ 341
imports_put_shares_groups ................................... 341
imports_put_shares_users ................................... 342
imports_put_syncs .......................................... 343
imports_put_syncs_archive .................................................. 348
jobs_delete_projects .......................................................... 350
jobs_delete_runs ............................................................... 351
jobs_deleteShares_groups .................................................... 351
jobs_deleteShares_users ...................................................... 352
jobs_get ........................................................................ 352
jobs_get_runs ................................................................. 353
jobs_list ........................................................................ 354
jobs_list_children ............................................................ 356
jobs_listParents ............................................................... 357
jobs_list_projects ............................................................. 358
jobs_list_runs ................................................................. 359
jobs_list_runs_logs ............................................................ 360
jobs_list_runs_outputs ....................................................... 361
jobs_list_shares ............................................................... 362
jobs_list_workflows ......................................................... 363
jobs_post_runs ................................................................. 364
jobs_post_trigger_email ...................................................... 365
jobs_put_archive .............................................................. 365
jobs_put_projects .............................................................. 367
jobs_putShares_groups ........................................................ 367
jobs_putShares_users .......................................................... 368
json_values_deleteShares_groups .......................................... 369
json_values_deleteShares_users ............................................. 370
json_values_get ................................................................. 370
json_values_list_shares ...................................................... 371
json_values_patch .............................................................. 372
json_values_post .............................................................. 372
json_values_putShares_groups .............................................. 373
json_values_putShares_users ................................................. 374
match_targets_deleteShares_groups ...................................... 375
match_targets_deleteShares_users ....................................... 376
match_targets_get ............................................................. 376
match_targets_list ............................................................ 377
match_targets_list_shares .................................................. 377
match_targets_patch .......................................................... 378
match_targets_post ............................................................ 379
match_targets_put_archive .................................................. 380
match_targets_putShares_groups .......................................... 380
match_targets_putShares_users ............................................. 381
media_delete_optimizations_runs ........................................ 382
media_delete_optimizationsShares_groups .............................. 383
media_delete_optimizationsShares_users ................................ 383
media_delete_ratecardsShares_groups ................................... 384
media_delete_ratecardsShares_users ..................................... 384
media_delete_spot_ordersShares_groups ................................. 385
media_delete_spot_ordersShares_users .................................. 385
media_get_optimizations ..................................................... 386
media_get_optimizations_runs ................................. 387
media_get_ratecards ........................................ 388
media_get_spot_orders ........................................ 389
media_list_dmas .............................................. 389
media_list_optimizations ...................................... 390
media_list_optimizations_runs ............................... 391
media_list_optimizations_runs_logs ........................ 392
media_list_optimizations_shares ............................. 393
media_list_ratecards ......................................... 394
media_list_ratecards_shares ................................. 394
media_list_spot_orders ...................................... 395
media_list_spot_orders_shares .............................. 396
media_list_targets ............................................ 397
media_patch_optimizations ................................... 397
media_patch_ratecards ....................................... 400
media_post_optimizations ..................................... 401
media_post_optimizations_clone ............................. 403
media_post_optimizations_runs .............................. 405
media_post_ratecards ......................................... 405
media_post_spot_orders ...................................... 406
media_put_optimizations_archive ............................ 407
media_put_optimizations_shares_groups ..................... 408
media_put_optimizations_shares_users ...................... 409
media_put_ratecards ......................................... 410
media_put_ratecards_archive ................................ 411
media_put_ratecards_shares_groups ......................... 412
media_put_ratecards_shares_users ......................... 413
media_put_spot_orders ...................................... 414
media_put_spot_orders_archive .............................. 415
media_put_spot_orders_shares_groups ....................... 415
media_put_spot_orders_shares_users ....................... 416
models_delete_builds ........................................ 417
models_delete_projects ....................................... 418
models_delete_shares_groups ............................... 418
models_delete_shares_users ................................. 419
models_get ..................................................... 419
models_get_builds ............................................ 422
models_list ..................................................... 423
models_list_builds .......................................... 426
models_list_builds_logs ...................................... 428
models_list_projects ......................................... 428
models_list_schedules ....................................... 429
models_list_shares .......................................... 430
models_list_types ............................................ 431
models_put_archive ........................................... 431
models_put_projects .......................................... 434
models_put_shares_groups .................................... 435
models_put_shares_users .................................... 436
topics documented:

multipart_upload ........................................ 437
notebooks_delete ........................................ 437
notebooks_delete_deployments ......................... 438
notebooks_delete_projects ............................... 438
notebooks_delete_shares_groups ......................... 439
notebooks_delete_shares_users ......................... 439
notebooks_get ........................................... 440
notebooks_get_deployments .............................. 442
notebooks_get_git_commits .............................. 443
notebooks_list ........................................... 443
notebooks_list_deployments ............................ 445
notebooks_list_deployments_logs ....................... 446
notebooks_list_git ...................................... 447
notebooks_list_git_commits ............................. 448
notebooks_list_projects ................................. 448
notebooks_list_shares ................................. 449
notebooks_list_update_links ............................ 450
notebooks_patch ......................................... 451
notebooks_post .......................................... 453
notebooks_post_clone ................................... 456
notebooks_post_deployments ............................ 458
notebooks_post_git_commits ............................ 459
notebooks_put ........................................... 460
notebooks_put_archive .................................. 463
notebooks_put_git ....................................... 465
notebooks_put_projects .................................. 466
notebooks_put_shares_groups ............................ 466
notebooks_put_shares_users ............................. 467
notifications_list ....................................... 468
ontology_list ............................................ 469
plot.civis_ml_classifier ................................ 470
plot.civis_ml_regressor ................................ 470
predictions_get ......................................... 471
predictions_list ......................................... 472
predictions_list_schedules .............................. 473
print.civis_api .......................................... 474
projects_delete ......................................... 474
projects_delete_parent_projects ....................... 475
projects_delete_shares_groups ......................... 475
projects_delete_shares_users ......................... 476
projects_get ............................................ 476
projects_list ............................................ 480
projects_list_parent_projects ......................... 481
projects_list_shares .................................... 482
projects_post ........................................... 483
projects_put ............................................ 487
projects_put_archive ................................... 491
projects_put_parent_projects ......................... 495
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>projects_put_shares_groups</td>
<td>496</td>
</tr>
<tr>
<td>projects_put_shares_users</td>
<td>497</td>
</tr>
<tr>
<td>publish_addin</td>
<td>498</td>
</tr>
<tr>
<td>publish_html</td>
<td>499</td>
</tr>
<tr>
<td>publish_rmd</td>
<td></td>
</tr>
<tr>
<td>queries_delete_runs</td>
<td>501</td>
</tr>
<tr>
<td>queries_get</td>
<td>501</td>
</tr>
<tr>
<td>queries_get_runs</td>
<td>502</td>
</tr>
<tr>
<td>queries_list</td>
<td>503</td>
</tr>
<tr>
<td>queries_list_runs</td>
<td>504</td>
</tr>
<tr>
<td>queries_list_runs_logs</td>
<td>505</td>
</tr>
<tr>
<td>queries_post</td>
<td>506</td>
</tr>
<tr>
<td>queries_post_runs</td>
<td>508</td>
</tr>
<tr>
<td>queries_put_scripts</td>
<td>509</td>
</tr>
<tr>
<td>query_civis</td>
<td>510</td>
</tr>
<tr>
<td>query_civis_file</td>
<td>511</td>
</tr>
<tr>
<td>read_civis</td>
<td>513</td>
</tr>
<tr>
<td>refresh_table</td>
<td>515</td>
</tr>
<tr>
<td>remote_hosts_list</td>
<td>515</td>
</tr>
<tr>
<td>remote_hosts_list_data_sets</td>
<td>516</td>
</tr>
<tr>
<td>remote_hosts_post</td>
<td>517</td>
</tr>
<tr>
<td>remote_hosts_post_authenticate</td>
<td></td>
</tr>
<tr>
<td>reports_delete_grants</td>
<td>518</td>
</tr>
<tr>
<td>reports_delete_projects</td>
<td>519</td>
</tr>
<tr>
<td>reports_delete_services_projects</td>
<td>519</td>
</tr>
<tr>
<td>reports_delete_services_shares_groups</td>
<td>520</td>
</tr>
<tr>
<td>reports_delete_services_shares_users</td>
<td>520</td>
</tr>
<tr>
<td>reports_delete_shares_groups</td>
<td>521</td>
</tr>
<tr>
<td>reports_delete_shares_users</td>
<td>521</td>
</tr>
<tr>
<td>reports_get</td>
<td>522</td>
</tr>
<tr>
<td>reports_get_git_commits</td>
<td>523</td>
</tr>
<tr>
<td>reports_get_services</td>
<td>524</td>
</tr>
<tr>
<td>reports_list</td>
<td>525</td>
</tr>
<tr>
<td>reports_list_git</td>
<td>527</td>
</tr>
<tr>
<td>reports_list_git_commits</td>
<td>527</td>
</tr>
<tr>
<td>reports_list_projects</td>
<td>528</td>
</tr>
<tr>
<td>reports_list_services_projects</td>
<td>529</td>
</tr>
<tr>
<td>reports_list_services_shares</td>
<td>530</td>
</tr>
<tr>
<td>reports_list_shares</td>
<td>531</td>
</tr>
<tr>
<td>reports_patch</td>
<td>532</td>
</tr>
<tr>
<td>reports_patch_services</td>
<td>534</td>
</tr>
<tr>
<td>reports_post</td>
<td>535</td>
</tr>
<tr>
<td>reports_post_git_commits</td>
<td>537</td>
</tr>
<tr>
<td>reports_post_grants</td>
<td>537</td>
</tr>
<tr>
<td>reports_post_refresh</td>
<td>539</td>
</tr>
<tr>
<td>reports_post_services</td>
<td>540</td>
</tr>
<tr>
<td>reports_put_archive</td>
<td>541</td>
</tr>
<tr>
<td>reports_put_git</td>
<td>542</td>
</tr>
</tbody>
</table>
R topics documented:

reports_put_projects ................................................. 543
reports_put_services_projects ....................................... 544
reports_put_services_shares_groups ................................. 544
reports_put_services_shares_users ................................. 545
reports_put_shares_groups ........................................... 546
reports_put_shares_users ............................................ 547
run_civis ............................................................. 548
run_template .......................................................... 549
saml_service_providers_delete_shares_groups ..................... 550
saml_service_providers_delete_shares_users ...................... 551
saml_service_providers_list_shares ............................... 551
saml_service_providers_put_shares_groups ....................... 552
saml_service_providers_put_shares_users ......................... 553
scripts_delete ....................................................... 554
scripts_delete_containers ............................................ 555
scripts_delete_containers_projects ................................. 555
scripts_delete_containers_runs ..................................... 556
scripts_delete_containers_shares_groups ......................... 556
scripts_delete_containers_shares_users ........................... 557
scripts_delete_custom ................................................ 557
scripts_delete_custom_projects ..................................... 558
scripts_delete_custom_runs .......................................... 558
scripts_delete_custom_shares_groups ............................... 559
scripts_delete_custom_shares_users ............................... 559
scripts_delete_javascript ............................................. 560
scripts_delete_javascript_projects ................................. 560
scripts_delete_javascript_runs ..................................... 561
scripts_delete_javascript_shares_groups ......................... 561
scripts_delete_javascript_shares_users ........................... 562
scripts_delete_python3 .............................................. 562
scripts_delete_python3_projects ................................... 563
scripts_delete_python3_runs ....................................... 563
scripts_delete_python3_shares_groups ............................. 564
scripts_delete_python3_shares_users .............................. 564
scripts_delete_r ..................................................... 565
scripts_delete_r_projects ........................................... 565
scripts_delete_r_runs ................................................. 566
scripts_delete_r_shares_groups .................................... 566
scripts_delete_r_shares_users ..................................... 567
scripts_delete_sql .................................................... 567
scripts_delete_sql_projects ........................................ 568
scripts_delete_sql_runs ............................................. 568
scripts_delete_sql_shares_groups ................................ 569
scripts_delete_sql_shares_users .................................. 569
scripts_get .......................................................... 570
scripts_get_containers .............................................. 573
scripts_get_containers_runs ....................................... 576
scripts_get_custom .................................................... 577
scripts_get_custom_runs ................................................................. 580
scripts_get_javascript ................................................................. 580
scripts_get_javascript_git_commits ................................................ 583
scripts_get_javascript_runs .......................................................... 584
scripts_get_python3 ................................................................. 584
scripts_get_python3_git_commits .................................................... 587
scripts_get_python3_runs .............................................................. 588
scripts_get_r ............................................................................. 589
scripts_get_r_git_commits ............................................................. 592
scripts_get_r_runs ................................................................. 592
scripts_get_sql ................................................................. 593
scripts_get_sql_git_commits ....................................................... 596
scripts_get_sql_runs .............................................................. 597
scripts_list ................................................................. 598
scripts_list_containers_projects .................................................. 600
scripts_list_containers_runs .......................................................... 601
scripts_list_containers_runs_logs .................................................. 602
scripts_list_containers_runs_outputs .................................................. 602
scripts_list_containersShares ....................................................... 603
scripts_list_custom ............................................................. 604
scripts_list_custom_projects ....................................................... 606
scripts_list_custom_runs ............................................................. 607
scripts_list_custom_runs_logs ....................................................... 608
scripts_list_custom_runs_outputs .................................................... 609
scripts_list_customShares .......................................................... 610
scripts_list_history ............................................................. 611
scripts_list_javascript_git ............................................................. 611
scripts_list_javascript_git_commits .................................................. 612
scripts_list_javascript_projects ..................................................... 613
scripts_list_javascript_runs .......................................................... 614
scripts_list_javascript_runs_logs ..................................................... 615
scripts_list_javascript_runs_outputs .................................................. 615
scripts_list_javascriptShares ......................................................... 616
scripts_list_python3_git ............................................................. 617
scripts_list_python3_git_commits ..................................................... 618
scripts_list_python3_projects ....................................................... 619
scripts_list_python3_runs ............................................................. 620
scripts_list_python3_runs_logs ....................................................... 621
scripts_list_python3_runs_outputs .................................................... 621
scripts_list_python3Shares .......................................................... 622
scripts_list_r_git ................................................................. 623
scripts_list_r_git_commits ............................................................. 624
scripts_list_r_projects ............................................................. 625
scripts_list_r_runs ................................................................. 626
scripts_list_r_runs_logs .............................................................. 627
scripts_list_r_runs_outputs .......................................................... 627
scripts_list_rShares ................................................................. 628
scripts_list_sql_git ................................................................. 629
topics documented:

scripts_list_sql_git_commits ........................................ 630
scripts_list_sql_projects .......................................... 631
scripts_list_sql_runs ............................................. 632
scripts_list_sql_runs_logs ....................................... 633
scripts_list_sql_runs_outputs ................................... 633
scripts_list_sql_shares ........................................... 634
scripts_list_types ................................................ 635
scripts_patch ....................................................... 636
scripts_patch_containers ......................................... 640
scripts_patch_custom ............................................ 646
scripts_patch_javascript .......................................... 650
scripts_patch_python3 ............................................ 654
scripts_patch_r ................................................... 659
scripts_patch_sql ................................................ 664
scripts_post ......................................................... 669
scripts_post_cancel ............................................... 673
scripts_post_containers .......................................... 674
scripts_post_containers_clone ................................ 679
scripts_post_containers_runs ................................... 683
scripts_post_containers_runs_logs .............................. 684
scripts_post_containers_runs_outputs ......................... 685
scripts_post_custom ............................................... 685
scripts_post_custom_clone ...................................... 689
scripts_post_custom_runs ......................................... 693
scripts_post_custom_runs_outputs ................................ 693
scripts_post_javascript ........................................... 694
scripts_post_javascript_clone .................................. 698
scripts_post_javascript_git_commits ............................ 701
scripts_post_javascript_runs ...................................... 702
scripts_post_javascript_runs_outputs ......................... 703
scripts_post_python3 ............................................. 703
scripts_post_python3_clone ..................................... 709
scripts_post_python3_git_commits ................................ 712
scripts_post_python3_runs ....................................... 713
scripts_post_python3_runs_outputs ............................. 713
scripts_post_r ..................................................... 714
scripts_post_run .................................................. 719
scripts_post_r_clone ............................................. 720
scripts_post_r_git_commits ....................................... 723
scripts_post_r_runs ............................................... 724
scripts_post_r_runs_outputs ..................................... 724
scripts_post_sql .................................................. 725
scripts_post_sql_clone ........................................... 730
scripts_post_sql_git_commits .................................... 733
scripts_post_sql_runs ............................................. 734
scripts_put_containers ........................................... 735
scripts_put_containers_archive .................................. 741
scripts_put_containers_projects ................................. 744
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>scripts_put_containers_shares_groups</td>
<td>745</td>
</tr>
<tr>
<td>scripts_put_containers_shares_users</td>
<td>746</td>
</tr>
<tr>
<td>scripts_put_custom</td>
<td>747</td>
</tr>
<tr>
<td>scripts_put_custom_archive</td>
<td>751</td>
</tr>
<tr>
<td>scripts_put_custom_projects</td>
<td>754</td>
</tr>
<tr>
<td>scripts_put_custom_shares_groups</td>
<td>754</td>
</tr>
<tr>
<td>scripts_put_custom_shares_users</td>
<td>755</td>
</tr>
<tr>
<td>scripts_put_custom_archive</td>
<td>756</td>
</tr>
<tr>
<td>scripts_put_custom_projects</td>
<td>756</td>
</tr>
<tr>
<td>scripts_put_custom_shares_groups</td>
<td>756</td>
</tr>
<tr>
<td>scripts_put_custom_shares_users</td>
<td>756</td>
</tr>
<tr>
<td>scripts_put_javascript</td>
<td>756</td>
</tr>
<tr>
<td>scripts_put_javascript_archive</td>
<td>761</td>
</tr>
<tr>
<td>scripts_put_javascript_git</td>
<td>761</td>
</tr>
<tr>
<td>scripts_put_javascript_projects</td>
<td>765</td>
</tr>
<tr>
<td>scripts_put_javascript_shares_groups</td>
<td>765</td>
</tr>
<tr>
<td>scripts_put_javascript_shares_users</td>
<td>766</td>
</tr>
<tr>
<td>scripts_put_javascript_archive</td>
<td>773</td>
</tr>
<tr>
<td>scripts_put_python3_archive</td>
<td>773</td>
</tr>
<tr>
<td>scripts_put_python3_projects</td>
<td>777</td>
</tr>
<tr>
<td>scripts_put_python3_shares_groups</td>
<td>777</td>
</tr>
<tr>
<td>scripts_put_python3_shares_users</td>
<td>778</td>
</tr>
<tr>
<td>scripts_put_r</td>
<td>779</td>
</tr>
<tr>
<td>scripts_put_r_archive</td>
<td>785</td>
</tr>
<tr>
<td>scripts_put_r_git</td>
<td>785</td>
</tr>
<tr>
<td>scripts_put_r_projects</td>
<td>789</td>
</tr>
<tr>
<td>scripts_put_r_shares_groups</td>
<td>789</td>
</tr>
<tr>
<td>scripts_put_r_shares_users</td>
<td>790</td>
</tr>
<tr>
<td>scripts_put_sql</td>
<td>791</td>
</tr>
<tr>
<td>scripts_put_sql_archive</td>
<td>797</td>
</tr>
<tr>
<td>scripts_put_sql_git</td>
<td>800</td>
</tr>
<tr>
<td>scripts_put_sql_projects</td>
<td>801</td>
</tr>
<tr>
<td>scripts_put_sql_shares_groups</td>
<td>801</td>
</tr>
<tr>
<td>scripts_put_sql_shares_users</td>
<td>802</td>
</tr>
<tr>
<td>search_list</td>
<td>803</td>
</tr>
<tr>
<td>search_list_types</td>
<td>805</td>
</tr>
<tr>
<td>services_delete</td>
<td>805</td>
</tr>
<tr>
<td>services_delete_deployments</td>
<td>806</td>
</tr>
<tr>
<td>services_delete_projects</td>
<td>806</td>
</tr>
<tr>
<td>services_delete_shares_groups</td>
<td>807</td>
</tr>
<tr>
<td>services_delete_shares_users</td>
<td>807</td>
</tr>
<tr>
<td>servicesdelete_tokens</td>
<td>808</td>
</tr>
<tr>
<td>services_get</td>
<td>808</td>
</tr>
<tr>
<td>services_get_deployments</td>
<td>811</td>
</tr>
<tr>
<td>services_list</td>
<td>812</td>
</tr>
<tr>
<td>services_list_deployments</td>
<td>814</td>
</tr>
<tr>
<td>services_list_deployments_logs</td>
<td>815</td>
</tr>
<tr>
<td>services_list_projects</td>
<td>816</td>
</tr>
<tr>
<td>services_list_shares</td>
<td>817</td>
</tr>
<tr>
<td>services_list_tokens</td>
<td>818</td>
</tr>
<tr>
<td>services_patch</td>
<td>819</td>
</tr>
</tbody>
</table>
R topics documented:

services_post ...................................................... 823
services_post_clone .............................................. 827
services_post_deployments ..................................... 829
services_post_redeploy ......................................... 830
services_post_tokens ............................................ 831
services_put ....................................................... 832
services_put_archive ........................................... 836
services_put_projects ........................................... 839
services_put_shares_groups .................................... 839
services_put_shares_users ..................................... 840
sql ................................................................. 841
stash_local_dataframe .......................................... 842
storage_hosts_delete ............................................ 842
storage_hosts_delete_shares_groups ......................... 843
storage_hosts_delete_shares_users ......................... 843
storage_hosts_get ............................................... 844
storage_hosts_list .............................................. 844
storage_hosts_list_shares ...................................... 845
storage_hosts_patch ............................................. 846
storage_hosts_post ............................................... 847
storage_hosts_put ............................................... 848
storage_hosts_put_shares_groups ......................... 849
storage_hosts_put_shares_users ............................ 850
tables_delete_projects ......................................... 851
tables_get ........................................................ 851
tables_get_enhancements_cass_ncoa ....................... 854
tables_get_enhancements_geocodings ...................... 855
tables_list ........................................................ 856
tables_list_columns ............................................. 857
tables_list_projects .......................................... 859
tables_patch ....................................................... 860
tables_post_enhancements_cass_ncoa ....................... 861
tables_post_enhancements_geocodings ..................... 862
tables_post_refresh ............................................ 863
tables_post_scan ................................................ 866
tables_put_projects ............................................. 866
templates_delete_reports ............................... 867
templates_delete_reports_shares_groups ................. 867
templates_delete_reports_shares_users ............... 868
templates_delete_scripts .................................... 868
templates_delete_scripts_projects ...................... 869
templates_delete_scripts_shares_groups ............... 869
templates_delete_scripts_shares_users ............... 870
templates_get_reports ......................................... 870
templates_get_scripts ......................................... 871
templates_list_reports ......................................... 872
templates_list_reports_shares ................................ 873
templates_list_scripts ......................................... 874
templates_list_scripts_projects ........................................ 875
templates_list_scripts_shares ........................................ 876
templates_patch_reports ............................................... 877
templates_patch_scripts ............................................... 878
templates_post_reports ............................................... 879
templates_post_reports_review ..................................... 881
templates_post_scripts ............................................... 882
templates_post_scripts_review ..................................... 883
templates_put_reports ............................................... 884
templates_put_reports_shares_groups ................................ 885
templates_put_reports_shares_users ................................ 886
templates_put_scripts ............................................... 887
templates_put_scripts_projects ..................................... 888
templates_put_scripts_shares_groups ................................ 889
templates_put_scripts_shares_users ................................ 890
transfer_table .......................................................... 891
users_delete_api_keys .................................................. 892
users_delete_me_favorites ............................................. 893
users_get ............................................................... 894
users_get_api_keys ...................................................... 895
users_list ............................................................... 896
users_list_api_keys ...................................................... 897
users_list_me ............................................................ 898
users_list_me_favorites .................................................. 899
users_list_me_ui .......................................................... 900
users_patch .............................................................. 901
users_patch_me ........................................................... 903
users_post ............................................................... 907
users_post_api_keys ...................................................... 909
users_post_me_favorites ................................................. 910
workflows_delete_projects ............................................ 911
workflows_delete_shares_groups ..................................... 911
workflows_delete_shares_users ...................................... 912
workflows_get ............................................................ 912
workflows_get_executions ............................................. 914
workflows_get_executions_tasks .................................... 915
workflows_get_git_commits ............................................ 916
workflows_list ............................................................ 917
workflows_list_executions ............................................. 918
workflows_list_git ....................................................... 919
workflows_list_git_commits .......................................... 920
workflows_list_projects ............................................... 921
workflows_list_shares .................................................. 922
workflows_patch .......................................................... 923
workflows_post ........................................................... 925
workflows_post_clone ................................................... 928
workflows_post_executions ............................................. 929
workflows_post_executions_cancel .................................. 931
**admin_delete_announcements**

*Delete an announcement*

**Description**

Delete an announcement

**Usage**

```
admin_delete_announcements(id)
```

**Arguments**

- **id**
  
  integer required. The ID of this announcement

**Value**

- An empty HTTP response

---

**admin_get_announcements**

*Get a particular announcement*

**Description**

Get a particular announcement

**Usage**

```
admin_get_announcements(id)
```
admin_list_announcements

Arguments

id integer required. The ID of this announcement

Value

A list containing the following elements:

id integer, The ID of this announcement
subject string, The subject of this announcement.
body string, The body of this announcement.
releasedAt string, The date and time this announcement was released.
createdAt string,
updatedAt string,

Description

List announcements

Usage

admin_list_announcements(
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)

Arguments

limit integer optional. Number of results to return. Defaults to 10. Maximum allowed is 50.
page_num integer optional. Page number of the results to return. Defaults to the first page, 1.
order string optional. The field on which to order the result set. Defaults to released_at. Must be one of: released_at.
order_dir string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.
Value

An array containing the following fields:

- **id** integer, The ID of this announcement.
- **subject** string, The subject of this announcement.
- **body** string, The body of this announcement.
- **releasedAt** string, The date and time this announcement was released.
- **createdAt** string,
- **updatedAt** string,

admin_list_organizations

List organizations

Description

List organizations

Usage

admin_list_organizations(status = NULL, org_type = NULL)

Arguments

- **status** array optional. The status of the organization (active/trial/inactive).
- **org_type** array optional. The organization type (platform/ads/survey_vendor/other).

Value

An array containing the following fields:

- **id** integer, The ID of this organization.
- **name** string, The name of this organization.
- **slug** string, The slug of this organization.
- **accountManagerId** integer, The user ID of the Account Manager.
- **csSpecialistId** integer, The user ID of the Client Success Specialist.
- **status** string, The status of the organization (active/trial/inactive).
- **orgType** string, The organization type (platform/ads/survey_vendor/other).
- **customBranding** string, The custom branding settings.
- **contractSize** integer, The monthly contract size.
- **maxAnalystUsers** integer, The max number of full platform users for the org.
admin_patch_announcements

maxReportUsers integer, The max number of report-only platform users for the org.
vertical string, The business vertical that the organization belongs to.
csMetadata string, Additional metadata about the organization in JSON format.
removeFooterInEmails boolean, If true, emails sent by platform will not include Civis text.
salesforceAccountId string, The SalesForce Account ID for this organization.
advancedSettings list, A list containing the following elements:
  • dedicatedDjPoolEnabled boolean, If true, the Organization has a dedicated delayed jobs pool. Defaults to false.
tableauRefreshHistory array, The number of tableau refreshes used this month.

admin_patch_announcements

Edit an announcement

Description
Edit an announcement

Usage
admin_patch_announcements(id, subject = NULL, body = NULL, released_at = NULL)

Arguments
id integer required. The ID of this announcement
subject string optional. The subject of this announcement.
body string optional. The body of this announcement.
released_at string optional. The date and time this announcement was released.

Value
A list containing the following elements:

id integer, The ID of this announcement
subject string, The subject of this announcement.
body string, The body of this announcement.
releasedAt string, The date and time this announcement was released.
createdAt string,
Description

Edit a theme

Usage

```python
admin_patch_themes(
    id,
    name = NULL,
    organization_ids = NULL,
    settings_json = NULL,
    logo_file_id = NULL
)
```

Arguments

- **id** integer required. The ID of this theme.
- **name** string optional. The name of this theme.
- **organization_ids** array optional. List of organization ID’s allowed to see this theme.
- **settings_json** string optional. The JSON-encoded theme configuration.
- **logo_file_id** string optional. The ID of the logo image file.

Value

A list containing the following elements:

- **id** integer, The ID of this theme.
- **name** string, The name of this theme.
- **organizationIds** array, List of organization ID’s allowed to use this theme.
- **settings** string, The theme configuration object.
- **logoFile** list, A list containing the following elements:
  - **id** integer, The ID of the logo image file.
  - **downloadUrl** string, The URL of the logo image file.
- **createdAt** string,
- **updatedAt** string,
admin_post_announcements

Post an announcement

Description
Post an announcement

Usage
admin_post_announcements(subject, body, released_at = NULL)

Arguments
- subject: string required. The subject of this announcement.
- body: string required. The body of this announcement.
- released_at: string optional. The date and time this announcement was released.

Value
A list containing the following elements:
- id: integer, The ID of this announcement
- subject: string, The subject of this announcement.
- body: string, The body of this announcement.
- releasedAt: string, The date and time this announcement was released.
- createdAt: string,
- updatedAt: string,

admin_post_themes
Create a theme

Description
Create a theme

Usage
admin_post_themes(
    name,
    settings_json,
    organization_ids = NULL,
    logo_file_id = NULL
)
Arguments

name: string required. The name of this theme.
settings_json: string required. The JSON-encoded theme configuration.
organization_ids: array optional. List of organization ID's allowed to see this theme.
logo_file_id: string optional. The ID of the logo image file.

Value

A list containing the following elements:

- id: integer, The ID of this theme.
- name: string, The name of this theme.
- organizationIds: array, List of organization ID’s allowed to use this theme.
- settings: string, The theme configuration object.
- logoFile: list, A list containing the following elements:
  - downloadUrl: string, The URL of the logo image file.
- createdAt: string,
- updatedAt: string,

aliases_delete

Delete an alias

Description

Delete an alias

Usage

aliases_delete(id)

Arguments

id: integer required. The id of the Alias object.

Value

An empty HTTP response
aliases_delete_shares_groups

*Revoke the permissions a group has on this object*

**Description**

Revoke the permissions a group has on this object

**Usage**

`aliases_delete_shares_groups(id, group_id)`

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of the resource that is shared.</td>
</tr>
<tr>
<td>group_id</td>
<td>integer required. The ID of the group.</td>
</tr>
</tbody>
</table>

**Value**

An empty HTTP response

---

aliases_delete_shares_users

*Revoke the permissions a user has on this object*

**Description**

Revoke the permissions a user has on this object

**Usage**

`aliases_delete_shares_users(id, user_id)`

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of the resource that is shared.</td>
</tr>
<tr>
<td>user_id</td>
<td>integer required. The ID of the user.</td>
</tr>
</tbody>
</table>

**Value**

An empty HTTP response
aliases_get_object_type

Get details about an alias within an FCO type

Description
Get details about an alias within an FCO type

Usage
aliases_get_object_type(object_type, alias)
**Arguments**

- **object_type** : string required. The type of the object. Valid types include: model, cass_ncoa, container_script, gdoc_export, geocode, media_optimizer, python_script, r_script, salesforce_export, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.

- **alias** : string required. The alias of the object

**Value**

A list containing the following elements:

- **id** : integer, The id of the Alias object.
- **objectId** : integer, The id of the object
- **objectType** : string, The type of the object. Valid types include: model, cass_ncoa, container_script, gdoc_export, geocode, media_optimizer, python_script, r_script, salesforce_export, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.
- **alias** : string, The alias of the object
- **userId** : integer, The id of the user who created the alias
- **displayName** : string, The display name of the Alias object. Defaults to object name if not provided.

---

**aliases_list**

---

**Description**

List Aliases

**Usage**

```python
aliases_list(
    object_type = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```

**Arguments**

aliases_list_shares

**Limit**
integer optional. Number of results to return. Defaults to 50. Maximum allowed is 1000.

**Page_num**
integer optional. Page number of the results to return. Defaults to the first page, 1.

**Order**
string optional. The field on which to order the result set. Defaults to id. Must be one of: id, object_type.

**Order_dir**
string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

### Value
An array containing the following fields:

- **id** integer, The id of the Alias object.
- **objectId** integer, The id of the object
- **objectType** string, The type of the object. Valid types include: model, cass_ncoa, container_script, gdoc_export, geocode, media_optimizer, python_script, r_script, salesforce_export, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.
- **alias** string, The alias of the object
- **userId** integer, The id of the user who created the alias
- **displayName** string, The display name of the Alias object. Defaults to object name if not provided.

---

**aliases_list_shares** List users and groups permissioned on this object

**Description**
List users and groups permissioned on this object

**Usage**
aliases_list_shares(id)

**Arguments**
- **id** integer required. The ID of the resource that is shared.
Value

An array containing the following fields:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,
- **writers** list, A list containing the following elements:
  - users array,
  - groups array,
- **owners** list, A list containing the following elements:
  - users array,
  - groups array,
- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

aliases_patch

*Update some attributes of this Alias*

Description

Update some attributes of this Alias

Usage

```python
aliases_patch(
    id,
    object_id = NULL,
    object_type = NULL,
    alias = NULL,
    display_name = NULL
)
```

Arguments

- **id** integer required. The id of the Alias object.
- **object_id** integer optional. The id of the object
- **object_type** string optional. The type of the object. Valid types include: model, cass_ncoa, container_script, gdoc_export, geocode, media_optimizer, python_script, r_script, salesforce_export, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.
- **alias** string optional. The alias of the object
- **display_name** string optional. The display name of the Alias object. Defaults to object name if not provided.
Value

A list containing the following elements:

- id: integer, The id of the Alias object.
- objectId: integer, The id of the object
- objectType: string, The type of the object. Valid types include: model, cass_ncoa, container_script, gdoc_export, geocode, media_optimizer, python_script, r_script, salesforce_export, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.
- alias: string, The alias of the object
- userId: integer, The id of the user who created the alias
- displayName: string, The display name of the Alias object. Defaults to object name if not provided.

aliases_post

Create an Alias

Description

Create an Alias

Usage

aliases_post(object_id, object_type, alias, display_name = NULL)

Arguments

- object_id: integer required. The id of the object
- object_type: string required. The type of the object. Valid types include: model, cass_ncoa, container_script, gdoc_export, geocode, media_optimizer, python_script, r_script, salesforce_export, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.
- alias: string required. The alias of the object
- display_name: string optional. The display name of the Alias object. Defaults to object name if not provided.

Value

A list containing the following elements:

- id: integer, The id of the Alias object.
- objectId: integer, The id of the object
aliases_put

objectType string, The type of the object. Valid types include: model, cass_ncoa, container_script, gdoc_export, geocode, media_optimizer, python_script, r_script, salesforce_export, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.
alias string, The alias of the object
userId integer, The id of the user who created the alias
displayName string, The display name of the Alias object. Defaults to object name if not provided.

aliases_put

Replace all attributes of this Alias

Description
Replace all attributes of this Alias

Usage
aliases_put(id, object_id, object_type, alias, display_name = NULL)

Arguments
id integer required. The id of the Alias object.
object_id integer required. The id of the object
object_type string required. The type of the object. Valid types include: model, cass_ncoa, container_script, gdoc_export, geocode, media_optimizer, python_script, r_script, salesforce_export, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.
alias string required. The alias of the object
display_name string optional. The display name of the Alias object. Defaults to object name if not provided.

Value
A list containing the following elements:
id integer, The id of the Alias object.
objectId integer, The id of the object
objectType string, The type of the object. Valid types include: model, cass_ncoa, container_script, gdoc_export, geocode, media_optimizer, python_script, r_script, salesforce_export, javascript_script, sql_script, project, notebook, workflow, template_script, template_report, service, report, tableau and service_report.
alias string, The alias of the object
userId integer, The id of the user who created the alias
displayName string, The display name of the Alias object. Defaults to object name if not provided.
aliases_put_shares_groups

*Set the permissions groups has on this object*

**Description**

Set the permissions groups has on this object

**Usage**

```python
aliases_put_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```

**Arguments**

- `id` integer required. The ID of the resource that is shared.
- `group_ids` array required. An array of one or more group IDs.
- `permission_level` string required. Options are: "read", "write", or "manage".
- `share_email_body` string optional. Custom body text for e-mail sent on a share.
- `send_shared_email` boolean optional. Send email to the recipients of a share.

**Value**

A list containing the following elements:

- `readers` list, A list containing the following elements:
  - users array,
  - groups array,
- `writers` list, A list containing the following elements:
  - users array,
  - groups array,
- `owners` list, A list containing the following elements:
  - users array,
  - groups array,
- `totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
set_permissions_users

Set the permissions users have on this object

Description

Set the permissions users have on this object

Usage

aliases_put_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)

Arguments

id
integer required. The ID of the resource that is shared.

user_ids
array required. An array of one or more user IDs.

permission_level
string required. Options are: "read", "write", or "manage".

share_email_body
string optional. Custom body text for e-mail sent on a share.

send_shared_email
boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers
list, A list containing the following elements:
  • users array,
  • groups array,

writers
list, A list containing the following elements:
  • users array,
  • groups array,

owners
list, A list containing the following elements:
  • users array,
• groups array,

  totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

  totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

--

announcements_list List announcements

Description

List announcements

Usage

  announcements_list(
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
  )

Arguments

  limit integer optional. Number of results to return. Defaults to 10. Maximum allowed is 50.

  page_num integer optional. Page number of the results to return. Defaults to the first page, 1.

  order string optional. The field on which to order the result set. Defaults to released_at. Must be one of: released_at.

  order_dir string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

  id integer, The ID of this announcement

  subject string, The subject of this announcement.

  body string, The body of this announcement.

  releasedAt string, The date and time this announcement was released.

  createdAt string,

  updatedAt string,
**Description**

Remove an App Instance from a project

**Usage**

```python
apps_delete_instances_projects(id, project_id, slug)
```

**Arguments**

- `id` (integer) required. The ID of the App Instance.
- `project_id` (integer) required. The ID of the project.
- `slug` (string) required. The slug for the application.

**Value**

An empty HTTP response

---

**Description**

Revoke the permissions a group has on this object

**Usage**

```python
apps_delete_instances_shares_groups(slug, id, group_id)
```

**Arguments**

- `slug` (string) required. The slug for the application.
- `id` (integer) required. The ID of the resource that is shared.
- `group_id` (integer) required. The ID of the group.

**Value**

An empty HTTP response
**apps_delete_releases_shares_groups**

Revoke the permissions a group has on this object

**Description**

Revoke the permissions a group has on this object

**Usage**

```
apps_delete_releases_shares_groups(slug, id, group_id)
```

**Arguments**

- `slug` string required. The slug for the application.
- `id` integer required. The ID of the resource that is shared.
- `group_id` integer required. The ID of the group.

**Value**

An empty HTTP response

---

**apps_delete_instances_shares_users**

Revoke the permissions a user has on this object

**Description**

Revoke the permissions a user has on this object

**Usage**

```
apps_delete_instances_shares_users(slug, id, user_id)
```

**Arguments**

- `slug` string required. The slug for the application.
- `id` integer required. The ID of the resource that is shared.
- `user_id` integer required. The ID of the user.

**Value**

An empty HTTP response
**Description**

Revoke the permissions a user has on this object

**Usage**

```python
apps_delete_releases_shares_users(slug, id, user_id)
```

**Arguments**

- `slug`  
  string required. The slug for the application.
- `id`  
  integer required. The ID of the resource that is shared.
- `user_id`  
  integer required. The ID of the user.

**Value**

An empty HTTP response

---

**Description**

List details of a Decision Application

**Usage**

```python
apps_get(slug)
```

**Arguments**

- `slug`  
  string required. The slug for the application.
Value
A list containing the following elements:

- **slug** string, The slug for the application.
- **id** integer, The unique id of the application.
- **instanceName** string, A word that describes an instance of this app.
- **name** string, The name of the application.
- **currentRelease** list, A list containing the following elements:
  - id integer, The unique id of the release.
  - appId integer, The id of the app the release belongs to.
  - reportTemplateId integer, ID of the report template for this release.
  - resources object, A hash of resources associated with this release.
  - archived string, The archival status of the requested item(s).
- **features** list, App features.

---

apps_get_instances

Return a given app instance

Description
Return a given app instance

Usage

apps_get_instances(id, slug)

Arguments

- **id** integer required. The unique id of the instance.
- **slug** string required. The slug for the application.

Value

A list containing the following elements:

- **id** integer, The unique id of the instance.
- **name** string, The name of the instance.
- **appReleaseId** integer, The id of the app release the instance belongs to.
- **reportId** integer, The id of the report the instance belongs to.
- **createdAt** string, The time the instance was created at.
- **user** list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

projectId integer, The id of the project collecting all the items that belong to this app instance.

authCodeUrl string,

apiKey string, A Civis API key that can be used by this app instance.

archived string, The archival status of the requested item(s).

---

**apps_get_releases**  
*Return a given app release*

---

**Description**

Return a given app release

**Usage**

`apps_get_releases(id, slug)`

**Arguments**

- **id** integer required. The unique id of the release.
- **slug** string required. The slug for the application.

**Value**

A list containing the following elements:

- **id** integer, The unique id of the release.
- **appId** integer, The id of the app the release belongs to.
- **reportTemplateId** integer, ID of the report template for this release.
- **resources** list, A hash of resources associated with this release.
- **archived** string, The archival status of the requested item(s).
**apps_list**

*List apps*

**Description**

List apps

**Usage**

apps_list()

**Value**

An array containing the following fields:

- **slug**: string, The slug for the application.
- **id**: integer, The unique id of the application.
- **instanceName**: string, A word that describes an instance of this app.
- **name**: string, The name of the application.

---

**apps_list_instances**

*List the instances of a Decision Application*

**Description**

List the instances of a Decision Application

**Usage**

apps_list_instances(
    slug,
    archived = NULL,
    app_release_id = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
**Arguments**

- **slug**
  - string required. The slug for the application.
- **archived**
  - string optional. The archival status of the requested item(s).
- **app_release_id**
  - integer optional. If supplied, return only instances matching this release.
- **limit**
  - integer optional. Number of results to return. Defaults to its maximum of 50.
- **page_num**
  - integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order**
  - string optional. The field on which to order the result set. Defaults to id. Must be one of: id, created_at.
- **order_dir**
  - string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

**Value**

An array containing the following fields:

- **id**
  - integer, The unique id of the instance.
- **name**
  - string, The name of the instance.
- **appReleaseId**
  - integer, The id of the app release the instance belongs to.
- **reportId**
  - integer, The id of the report the instance belongs to.
- **createdAt**
  - string, The time the instance was created at.
- **user**
  - list, A list containing the following elements:
    - **id** integer, The ID of this user.
    - **name** string, This user’s name.
    - **username** string, This user’s username.
    - **initials** string, This user’s initials.
    - **online** boolean, Whether this user is online.
- **projectId**
  - integer, The id of the project collecting all the items that belong to this app instance.
- **archived**
  - string, The archival status of the requested item(s).

---

**apps_list_instances_projects**

*List the projects an App Instance belongs to*

**Description**

List the projects an App Instance belongs to

**Usage**

```python
apps_list_instances_projects(id, slug, hidden = NULL)
```
Args

**id**
integer required. The ID of the App Instance.

**slug**
string required. The slug for the application.

**hidden**
boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

- **id** integer, The ID for this project.
- **author** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **name** string, The name of this project.
- **description** string, A description of the project.
- **users** array, An array containing the following fields:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **autoShare** boolean,
- **createdAt** string,
- **updatedAt** string,
- **archived** string, The archival status of the requested item(s).

apps_list_instances_shares

*List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```
apps_list_instances_shares(slug, id)
```
**apps_list_releases**

**Arguments**

- **slug** string required. The slug for the application.
- **id** integer required. The ID of the resource that is shared.

**Value**

An array containing the following fields:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,

- **writers** list, A list containing the following elements:
  - users array,
  - groups array,

- **owners** list, A list containing the following elements:
  - users array,
  - groups array,

- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**Description**

List the releases of a particular Decision Application

**Usage**

```python
apps_list_releases(
    slug,
    archived = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>slug</td>
<td>string</td>
<td>required. The slug for the application.</td>
</tr>
<tr>
<td>archived</td>
<td>string</td>
<td>optional. The archival status of the requested item(s).</td>
</tr>
<tr>
<td>limit</td>
<td>integer</td>
<td>optional. Number of results to return. Defaults to its maximum of 50.</td>
</tr>
<tr>
<td>page_num</td>
<td>integer</td>
<td>optional. Page number of the results to return. Defaults to the first page, 1.</td>
</tr>
<tr>
<td>order</td>
<td>string</td>
<td>optional. The field on which to order the result set. Defaults to id. Must be one of: id.</td>
</tr>
<tr>
<td>order_dir</td>
<td>string</td>
<td>optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.</td>
</tr>
</tbody>
</table>

Value

An array containing the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td>The unique id of the release.</td>
</tr>
<tr>
<td>appId</td>
<td>integer</td>
<td>The id of the app the release belongs to.</td>
</tr>
<tr>
<td>reportTemplateId</td>
<td>integer</td>
<td>ID of the report template for this release.</td>
</tr>
<tr>
<td>resources</td>
<td>list</td>
<td>A hash of resources associated with this release.</td>
</tr>
<tr>
<td>archived</td>
<td>string</td>
<td>The archival status of the requested item(s).</td>
</tr>
</tbody>
</table>

---

apps_list_releases_shares

*List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

apps_list_releases_shares(slug, id)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>slug</td>
<td>string</td>
<td>required. The slug for the application.</td>
</tr>
<tr>
<td>id</td>
<td>integer</td>
<td>required. The ID of the resource that is shared.</td>
</tr>
</tbody>
</table>
Value

An array containing the following fields:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,

- **writers** list, A list containing the following elements:
  - users array,
  - groups array,

- **owners** list, A list containing the following elements:
  - users array,
  - groups array,

- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**apps_patch_instances**  
*Update a given app instance*

---

**Description**

Update a given app instance

**Usage**

`apps_patch_instances(id, slug, name = NULL)`

**Arguments**

- **id** integer required. The unique id of the instance.
- **slug** string required. The slug for the application.
- **name** string optional. The name of the instance.

**Value**

A list containing the following elements:

- **id** integer, The unique id of the instance.
- **name** string, The name of the instance.
- **appReleaseId** integer, The id of the app release the instance belongs to.
- **reportId** integer, The id of the report the instance belongs to.
createdAt  
string, The time the instance was created at.

user  
list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

projectId  
integer, The id of the project collecting all the items that belong to this app instance.

authCodeUrl  
string,

apiKey  
string, A Civis API key that can be used by this app instance.

archived  
string, The archival status of the requested item(s).

---

**apps_patch_releases**  
*Update an existing Decision Application release*

**Description**

Update an existing Decision Application release

**Usage**

`apps_patch_releases(slug, id, report_template_id = NULL, resources = NULL)`

**Arguments**

- **slug**  
  string required. The slug for the application.

- **id**  
  integer required. The unique id of the release.

- **report_template_id**  
  integer optional. ID of the report template for this release.

- **resources**  
  list optional. A hash of resources associated with this release.

**Value**

A list containing the following elements:

- **id**  
  integer, The unique id of the release.

- **appId**  
  integer, The id of the app the release belongs to.

- **reportTemplateId**  
  integer, ID of the report template for this release.

- **resources**  
  list, A hash of resources associated with this release.

- **archived**  
  string, The archival status of the requested item(s).
Description
Create a new instance of an application of the given slug

Usage
apps_post_instances(slug, name = NULL)

Arguments
slug string required. The slug for the application.
name string optional. The name of the instance.

Value
A list containing the following elements:

id integer, The unique id of the instance.
name string, The name of the instance.
appReleaseId integer, The id of the app release the instance belongs to.
reportId integer, The id of the report the instance belongs to.
createdAt string, The time the instance was created at.
user list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
projectId integer, The id of the project collecting all the items that belong to this app instance.
authCodeUrl string,
apiKey string, A Civis API key that can be used by this app instance.
archived string, The archival status of the requested item(s).
apps_post_releases  
Create a new Decision Application release

Description
Create a new Decision Application release

Usage
apps_post_releases(slug, report_template_id, resources)

Arguments
slug  
string required. The slug for the application.

report_template_id  
integer required. ID of the report template for this release.

resources  
list required. A hash of resources associated with this release.

Value
A list containing the following elements:

id  
integer, The unique id of the release.

appId  
integer, The id of the app the release belongs to.

reportTemplateId  
integer, ID of the report template for this release.

resources  
list, A hash of resources associated with this release.

archived  
string, The archival status of the requested item(s).

apps_put_features  
Update the Decision Application features for a given organization

Description
Update the Decision Application features for a given organization

Usage
apps_put_features(slug, org, features)

Arguments
slug  
string required. The slug for the application.

org  
string required. Organization.

features  
list required. App features.
Value
A list containing the following elements:

- **slug** string, The slug for the application.
- **id** integer, The unique id of the application.
- **instanceName** string, A word that describes an instance of this app.
- **name** string, The name of the application.
- **currentRelease** list, A list containing the following elements:
  - **id** integer, The unique id of the release.
  - **appId** integer, The id of the app the release belongs to.
  - **reportTemplateId** integer, ID of the report template for this release.
  - **resources** object, A hash of resources associated with this release.
  - **archived** string, The archival status of the requested item(s).
- **features** list, App features.

Description
Update the archive status of this object

Usage
apps_put_instances_archive(id, slug, status)

Arguments
- **id** integer required. The ID of the object.
- **slug** string required. The slug for the application.
- **status** boolean required. The desired archived status of the object.

Value
A list containing the following elements:

- **id** integer, The unique id of the instance.
- **name** string, The name of the instance.
- **appReleaseId** integer, The id of the app release the instance belongs to.
- **reportId** integer, The id of the report the instance belongs to.
- **createdAt** string, The time the instance was created at.
- **user** list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

projectId integer, The id of the project collecting all the items that belong to this app instance.
authCodeUrl string,
apiKey string, A Civis API key that can be used by this app instance.
archived string, The archival status of the requested item(s).

 apps_put_instances_projects
 Add an App Instance to a project

Description

Add an App Instance to a project

Usage

apps_put_instances_projects(id, project_id, slug)

Arguments

id integer required. The ID of the App Instance.
project_id integer required. The ID of the project.
slug string required. The slug for the application.

Value

An empty HTTP response
Set the permissions groups has on this object

Usage

apps_put_instances_shares_groups(
    slug,
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)

Arguments

slug string required. The slug for the application.

id integer required. The ID of the resource that is shared.

group_ids array required. An array of one or more group IDs.

permission_level string required. Options are: "read", "write", or "manage".

share_email_body string optional. Custom body text for e-mail sent on a share.

send_shared_email boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers list, A list containing the following elements:
    • users array,
    • groups array,

writers list, A list containing the following elements:
    • users array,
    • groups array,

owners list, A list containing the following elements:
    • users array,
    • groups array,
totalUserShares
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

apps_put_instances_shares_users
Set the permissions users have on this object

---

**Description**

Set the permissions users have on this object

**Usage**

```python
apps_put_instances_shares_users(
    slug,
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```

**Arguments**

- **slug** string required. The slug for the application.
- **id** integer required. The ID of the resource that is shared.
- **user_ids** array required. An array of one or more user IDs.
- **permission_level** string required. Options are: "read", "write", or "manage".
- **share_email_body** string optional. Custom body text for e-mail sent on a share.
- **send_shared_email** boolean optional. Send email to the recipients of a share.

**Value**

A list containing the following elements:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,

- **writers** list, A list containing the following elements:
apps_put_releases_archive

- users array,
- groups array,

owners list, A list containing the following elements:
  - users array,
  - groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

apps_put_releases_archive

*Update the archive status of this object*

**Description**

Update the archive status of this object

**Usage**

apps_put_releases_archive(id, slug, status)

**Arguments**

- **id** integer required. The ID of the object.
- **slug** string required. The slug for the application.
- **status** boolean required. The desired archived status of the object.

**Value**

A list containing the following elements:

- **id** integer, The unique id of the release.
- **appId** integer, The id of the app the release belongs to.
- **reportTemplateId** integer, ID of the report template for this release.
- **resources** list, A hash of resources associated with this release.
- **archived** string, The archival status of the requested item(s).
apps_put_releases_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

apps_put_releases_shares_groups(
    slug,
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)

Arguments

slug       string required. The slug for the application.
id         integer required. The ID of the resource that is shared.
group_ids  array required. An array of one or more group IDs.
permission_level
            string required. Options are: "read", "write", or "manage".
share_email_body
            string optional. Custom body text for e-mail sent on a share.
send_shared_email
            boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers    list, A list containing the following elements:
            • users array,
            • groups array.

writers    list, A list containing the following elements:
            • users array,
            • groups array.

owners     list, A list containing the following elements:
            • users array,
            • groups array.
```markdown
**Description**

Set the permissions users have on this object

**Usage**

```python
apps_put_releases_shares_users(
    slug,
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```

**Arguments**

- **slug**
  - string required. The slug for the application.
- **id**
  - integer required. The ID of the resource that is shared.
- **user_ids**
  - array required. An array of one or more user IDs.
- **permission_level**
  - string required. Options are: "read", "write", or "manage".
- **share_email_body**
  - string optional. Custom body text for e-mail sent on a share.
- **send_shared_email**
  - boolean optional. Send email to the recipients of a share.

**Value**

A list containing the following elements:

- **readers**
  - list, A list containing the following elements:
    - users array,
    - groups array,

- **writers**
  - list, A list containing the following elements:
```

await

-(users array,
- groups array,

owners list, A list containing the following elements:
- users array,
- groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**await**  
*Call a function repeatedly until a status is reached.*

---

**Description**

`await` repeatedly calls a Civis API endpoint such as `scripts_get_sql_runs` that monitors the status of a script, job, import, or model. It blocks until the function returns a result with a successful or error status. If the script, job, import or model results in an error state, `await` throws an error with useful debugging information.

`await_all` is a vectorized version of `await`. It repeatedly calls a Civis API endpoint for all values of a vector, e.g. a vector of script, job, import, run, or model ids. It blocks until all calls have returned a result with a given status, and silently captures jobs that return errors.

**Usage**

```r
await(  
  f,  
  ...,  
  .status_key = "state",  
  .success_states = c("succeeded", "success"),  
  .error_states = c("failed", "cancelled"),  
  .timeout = NULL,  
  .interval =getOption("civis.default.polling_interval"),  
  .verbose = FALSE  
)

await_all(  
  f,  
  .x,  
  .y = NULL,  
  ...,  
  .status_key = "state",  
  .success_states = c("succeeded", "success"),  
```
await

`.error_states = c("failed", "cancelled"),
.timeout = NULL,
.interval = NULL,
.verbose = FALSE
``

Arguments

- **f**: function to be called repeatedly until a status is reached.
- **...**: arguments to `f`
- **.status_key**: The name of the element of the list returned by `f` containing the status. For most Civis API endpoints, this is the default, "state".
- **.success_states**: list of states indicating remote work has completed successfully. For most Civis API endpoints, this set of states is the default, c("succeeded","success").
- **.error_states**: list of states indicating remote work is in an error state. For most Civis API endpoints, this set of states is the default, c("failed","cancelled").
- **.timeout**: Number of seconds after which to timeout.
- **.interval**: The interval for retries (in seconds). If `NULL` (default), use exponentially increasing intervals with jitter (see 'Details')
- **.verbose**: Print the status of `f` at a given retry with the retry time (default `FALSE`)
- **.x**: a vector of values to be passed to `f`
- **.y**: a vector of values to be passed to `f` (default `NULL`)

Details

`await` and `await_all` can wrap Civis API endpoints in `generated_client.R`. The default values for `.status_key`, `.success_states`, and `.error_states` are suitable for most endpoints. The final status of `f` can be obtained using `get_status`.

If an error state is reached, `await` throws a `civis_await_error`. `await_all` silently captures and returns a `civis_await_error` for any job reaching an error state as an element in the list of results. If `.timeout` is specified and the job fails to reach a success state within the time limit, `await` throws a `civis_timeout_error`. Likewise, `await_all` throws a `civis_timeout_error` if all jobs fail to reach a success state within the time limit.

These errors can be caught using `try` or `tryCatch`. Useful debugging information can be returned using `get_error` and `fetch_logs`.

The set of possible states for jobs on Civis platform are: "succeeded", "success", "failed", "queued", "running", and "cancelled".

Unless `.interval` is specified, retries are attempted with exponentially increasing intervals using 
\[ .25 \times (1.2^i) + \text{runif}(1,0,.2), \]\nwhere `i` is the index of the current retry. Approximate intervals for a given number of retries are as follows:

- 1-5: .5s
- 6-10: 1-5s
- 11-19: 5-10s
• 20-29: 10s - 1m

The polling interval can be set to a fixed value globally with options("civis.default_polling_interval" = INTERVAL_IN_SECONDS).

Functions

• `await_all`: Call a function repeatedly for all values of a vector until all have reached a completed status

See Also

`get_status`, `get_error`, `fetch_logs`

Examples

## Not run:

```r
# Executing a query
q_id <- queries_post(db_id, query, n_rows, cred_id)["id"]
r <- await(queries_get, id = q_id)
get_status(r)

r <- tryCatch(await(queries_get, id = q_id), error = function(e) e)
get_error(r)

r <- try(await(queries_get, id = q_id))
get_error(r)

jobs <- c(1234, 5678)
runs <- c(1234, 5678)
rs <- await_all(scripts_get_r_runs, .x = jobs, .y = runs)
```

## End(Not run)

---

cancel

**Cancel the evaluation of a CivisFuture.**

Description

Cancel the evaluation of a CivisFuture.

Usage

cancel(future, ...)

Arguments

- `future`: CivisFuture object.
- `...`: unused for CivisFuture.
Evaluate an expression in Civis Platform

**Description**

Evaluate an expression in Civis Platform

**Usage**

CivisFuture(
    expr = NULL,
    envir = parent.frame(),
    substitute = FALSE,
    globals = TRUE,
    packages = NULL,
    lazy = FALSE,
    local = TRUE,
    gc = FALSE,
    earlySignal = FALSE,
    label = NULL,
    required_resources = list(cpu = 1024, memory = 2048, diskSpace = 4),
    docker_image_name = "civisanalytics/datascience-r",
    docker_image_tag = "latest",
    ...
)

## S3 method for class 'CivisFuture'
run(future, ...)

## S3 method for class 'CivisFuture'
result(future, ...)

## S3 method for class 'CivisFuture'
cancel(future, ...)

## S3 method for class 'CivisFuture'
resolved(future, ...)

## S3 method for class 'CivisFuture'
fetch_logs(object, ...)

**Arguments**

- **expr**  
  An R expression.

- **envir**  
  The environment from where global objects should be identified.

- **substitute**  
  If TRUE, argument expr is substitute():ed; otherwise not.
globals (optional) a logical, a character vector, or a named list to control how globals are handled. For details, see section ’Globals used by future expressions’ in the help for `future()`.

packages (optional) a character vector specifying packages to be attached in the R environment evaluating the future.

lazy If FALSE (default), the future is resolved eagerly (starting immediately), otherwise not.

local If TRUE, the expression is evaluated such that all assignments are done to local temporary environment, otherwise the assignments are done to the global environment of the R process evaluating the future.

gc If TRUE, the garbage collector run (in the process that evaluated the future) only after the value of the future is collected. Exactly when the values are collected may depend on various factors such as number of free workers and whether earlySignal is TRUE (more frequently) or FALSE (less frequently). Some types of futures ignore this argument.

earlySignal Specified whether conditions should be signaled as soon as possible or not.

label An optional character string label attached to the future.

required_resources resources, see `scripts_post_containers`

docker_image_name the image for the container script.

docker_image_tag the tag for the Docker image.

... arguments to `scripts_post_containers`

future CivisFuture object.

object CivisFuture

Value

A CivisFuture inheriting from `Future` that evaluates `expr` on the given container.

Methods (by generic)

- run: Run a CivisFuture
- result: Return the value of a CivisFuture
- cancel: Cancel a CivisFuture
- resolved: Check if a CivisFuture has resolved
- fetch_logs: Fetch logs from a CivisFuture
**civis_file**  
*A file in the Civis Platform*

**Description**

Use `civis_file` to use a file in the Civis Platform with `civis_ml`.

**Usage**

```
civis_file(file_id)
```

**Arguments**

- `file_id`  
  The id of a Civis file.

**Value**

A `civis_file`.

---

**civis_file_manifest**  
*A manifest file in the Civis Platform*

**Description**

Use `civis_file_manifest` to use a manifest file in the Civis Platform with `civis_ml`.

**Usage**

```
civis_file_manifest(file_id)
```

**Arguments**

- `file_id`  
  The id of a Civis file.

**Value**

A `civis_file_manifest`
civis_ml  

Interface for modeling in the Civis Platform

Description

An interface for training and scoring data on Civis Platform using a set of Scikit-Learn estimators.

Usage

civis_ml(
  x,
  dependent_variable,
  model_type,
  primary_key = NULL,
  excluded_columns = NULL,
  parameters = NULL,
  fit_params = NULL,
  cross_validation_parameters = NULL,
  calibration = NULL,
  oos_scores_table = NULL,
  oos_scores_db = NULL,
  oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
  model_name = NULL,
  cpu_requested = NULL,
  memory_requested = NULL,
  disk_requested = NULL,
  notifications = NULL,
  polling_interval = NULL,
  validation_data = c("train", "skip"),
  n_jobs = NULL,
  verbose = FALSE,
  civisml_version = "prod"
)

civis_ml_fetch_existing(model_id, run_id = NULL)

## S3 method for class 'civis_ml'
predict(
  object,
  newdata,
  primary_key = NA,
  output_table = NULL,
  output_db = NULL,
  if_output_exists = c("fail", "append", "drop", "truncate"),
  n_jobs = NULL,
  cpu_requested = NULL,
  memory_requested = NULL,
disk_requested = NULL,
polling_interval = NULL,
verbose = FALSE,
dvs_to_predict = NULL,
...
)

Arguments

x, newdata See the Data Sources section below.
dependent_variable
The dependent variable of the training dataset. For a multi-target problem, this
should be a vector of column names of dependent variables. Nulls in a single
dependent variable will automatically be dropped.
model_type The name of the CivisML workflow. See the Workflows section below.
primary_key Optional, the unique ID (primary key) of the training dataset. This will be used
to index the out-of-sample scores. In predict.civis_ml, the primary_key of
the training task is used by default primary_key = NA. Use primary_key = NULL
to explicitly indicate the data have no primary_key.
excluded_columns
Optional, a vector of columns which will be considered ineligible to be indepen-
dent variables.
parameters
Optional, parameters for the final stage estimator in a predefined model, e.g.
list(C = 2) for a "sparse_logistic" model.
fit_params
Optional, a mapping from parameter names in the model’s fit method to the
column names which hold the data, e.g. list(sample_weight = 'survey_weight_column').
cross_validation_parameters
Optional, parameter grid for learner parameters, e.g. list(n_estimators =
c(100,200,500),learning_rate = c(0.01,0.1),max_depth = c(2,3)) or "hyperband"
for supported models.
calibration
Optional, if not NULL, calibrate output probabilities with the selected method,
sigmoid, or isotonic. Valid only with classification models.
oos_scores_table
Optional, if provided, store out-of-sample predictions on training set data to this
Redshift "schema.tablename".
oos_scores_db
Optional, the name of the database where the oos_scores_table will be cre-
ated. If not provided, this will default to database_name.
oos_scores_if_exists
Optional, action to take if oos_scores_table already exists. One of "fail",
"append", "drop", or "truncate". The default is "fail".
model_name
Optional, the prefix of the Platform modeling jobs. It will have " Train" or "
Predict" added to become the Script title.
cpu_requested
Optional, the number of CPU shares requested in the Civis Platform for training
jobs or prediction child jobs. 1024 shares = 1 CPU.
memory_requested
Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.

disk_requested
Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.

notifications
Optional, model status notifications. See scripts_post_custom for further documentation about email and URL notification.

polling_interval
Check for job completion every this number of seconds.

validation_data
Optional, source for validation data. There are currently two options: train (the default), which uses training data for validation, and skip, which skips the validation step.

n_jobs
Number of concurrent Platform jobs to use for training and validation, or multi-file / large table prediction. Defaults to NULL, which allows CivisML to dynamically calculate an appropriate number of workers to use (in general, as many as possible without using all resources in the cluster).

verbose
Optional, If TRUE, supply debug outputs in Platform logs and make prediction child jobs visible.

civisml_version
Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production

model_id
The id of CivisML model built previously.

run_id
Optional, the id of a CivisML model run. If NULL, defaults to fetching the latest run.

object
A civis_ml object.

output_table
The table in which to put predictions.

output_db
The database containing output_table. If not provided, this will default to the database_name specified when the model was built.

if_output_exists
Action to take if the prediction table already exists. One of "fail", "append", "drop", or "truncate". The default is "fail".

dvs_to_predict
Optional, For scoring, this should be a vector of column names of dependent variables to include in the output table. It must be a subset of the dependent_variable vector provided for training. The scores for the returned subset will be identical to the scores which those outputs would have had if all outputs were written, but ignoring some of the model’s outputs will let predictions complete faster and use less disk space. If not provided, the entire model output will be written to the output table.

... Unused

Value
A civis_ml object, a list containing the following elements:
job | job metadata from `scripts_get_custom`.
run | run metadata from `scripts_get_custom_runs`.
outputs | CivisML metadata from `scripts_list_custom_runs_outputs` containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.
metrics | Parsed CivisML output from `metrics.json` containing metadata from validation. A list containing the following elements:
  - run list, metadata about the run.
  - data list, metadata about the training data.
  - model list, the fitted scikit-learn model with CV results.
  - metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
  - warnings list.
  - data_platform list, training data location.
model_info | Parsed CivisML output from `model_info.json` containing metadata from training. A list containing the following elements:
  - run list, metadata about the run.
  - data list, metadata about the training data.
  - model list, the fitted scikit-learn model.
  - metrics empty list.
  - warnings list.
  - data_platform list, training data location.

### CivisML Workflows

You can use the following pre-defined models with `civis_ml`. All models start by imputing missing values with the mean of non-null values in a column. The "sparse_*" models include a LASSO regression step (using `glmnet`) to do feature selection before passing data to the final model. In some models, CivisML uses default parameters from those in Scikit-Learn, as indicated in the "Altered Defaults" column. All models also have `random_state=42`.

Specific workflows can also be called directly using the R workflow functions.

<table>
<thead>
<tr>
<th>Name</th>
<th>R Workflow</th>
<th>Model Type</th>
<th>Altered Defaults</th>
</tr>
</thead>
<tbody>
<tr>
<td>sparse_logistic</td>
<td><code>civis_ml_sparse_logistic</code></td>
<td>classification</td>
<td></td>
</tr>
<tr>
<td>gradient_boosting_classifier</td>
<td><code>civis_ml_gradient_boosting_classifier</code></td>
<td>classification</td>
<td></td>
</tr>
<tr>
<td>random_forest_classifier</td>
<td><code>civis_ml_random_forest_classifier</code></td>
<td>classification</td>
<td></td>
</tr>
<tr>
<td>extra_trees_classifier</td>
<td><code>civis_ml_extra_trees_classifier</code></td>
<td>classification</td>
<td></td>
</tr>
<tr>
<td>multilayer_perceptron_classifier</td>
<td><code>civis_ml_multilayer_perceptron_classifier</code></td>
<td>classification</td>
<td></td>
</tr>
<tr>
<td>stacking_classifier</td>
<td><code>civis_ml_stacking_classifier</code></td>
<td>classification</td>
<td></td>
</tr>
<tr>
<td>sparse_linear_regressor</td>
<td><code>civis_ml_sparse_linear_regressor</code></td>
<td>regression</td>
<td></td>
</tr>
<tr>
<td>sparse_ridge_regressor</td>
<td><code>civis_ml_sparse_ridge_regressor</code></td>
<td>regression</td>
<td></td>
</tr>
<tr>
<td>gradient_boosting_regressor</td>
<td><code>civis_ml_gradient_boosting_regressor</code></td>
<td>regression</td>
<td></td>
</tr>
<tr>
<td>random_forest_regressor</td>
<td><code>civis_mlRandom_forest_regressor</code></td>
<td>regression</td>
<td></td>
</tr>
<tr>
<td>extra_trees_regressor</td>
<td><code>civis_ml_extra_trees_regressor</code></td>
<td>regression</td>
<td></td>
</tr>
<tr>
<td>multilayer_perceptron_regressor</td>
<td><code>civis_ml_multilayer_perceptron_regressor</code></td>
<td>regression</td>
<td></td>
</tr>
<tr>
<td>stacking_regressor</td>
<td><code>civis_ml_stacking_regressor</code></td>
<td>regression</td>
<td></td>
</tr>
</tbody>
</table>
Model names can be easily accessed using the global variables CIVIS_ML_REGRESSORS and CIVIS_ML_CLASSIFIERS.

Stacking

The "stacking_classifier" model stacks together the "gradient_boosting_classifier" and "random_forest_classifier" predefined models together with a glmnet.LogitNet(alpha=0,n_splits=4,max_iter=10000,tol=1e-5,scoring=Var.log_loss) and then the model predictions are combined using LogisticRegressionCV with penalty='l2' and tol=1e-08. The "stacking_regressor" works similarly, stacking together the "gradient_boosting_regressor" and "random_forest_regressor" models and a glmnet.ElasticNet(alpha=0,n_splits=4,max_iter=10000,tol=1e-5) combining them using NonNegativeLinearRegression. The estimators that are being stacked have the same names as the associated pre-defined models, and the meta-estimator steps are named "meta-estimator". Note that although default parameters are provided for multilayer perceptron models, it is highly recommended that multilayer perceptrons be run using hyperband.

Hyperparameter Tuning

You can tune hyperparameters using one of two methods: grid search or hyperband. CivisML will perform grid search if you pass a list of hyperparameters to the cross_validation_parameters parameter, where list elements are hyperparameter names, and the values are vectors of hyperparameter values to grid search over. You can run hyperparameter optimization in parallel by setting the n_jobs parameter to however many jobs you would like to run in parallel. By default, n_jobs is dynamically calculated based on the resources available on your cluster, such that a modeling job will never take up more than 90

Hyperband is an efficient approach to hyperparameter optimization, and recommended over grid search where possible. CivisML will perform hyperband optimization if you pass the string "hyperband" to cross_validation_parameters. Hyperband is currently only supported for the following models: "gradient_boosting_classifier", "random_forest_classifier", "extra_trees_classifier", "multilayer_perceptron_classifier", "stacking_classifier", "gradient_boosting_regressor", "random_forest_regressor", "extra_trees_regressor", "multilayer_perceptron_regressor", and "stacking_regressor". Hyperband cannot be used to tune GLMs. For this reason, preset GLMs do not have a hyperband option. Similarly, when cross_validation_parameters='hyperband' and the model is stacking_classifier or stacking_regressor, only the GBT and random forest steps of the stacker are tuned using hyperband. For the specific distributions used in the predefined hyperband models, see the detailed table in the Python client documentation.

Data Sources

For building models with civis_ml, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a data.frame resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling civis_ml:

data.frame civis_ml(x = df,...)
local csv file civis_ml(x = "path/to/data.csv",...)
file in Civis Platform civis_ml(x = civis_file(1234))
table in Civis Platform civis_ml(x = civis_table(table_name = "schema.table",database_name = "database"))
Out of sample scores

Model outputs will always contain out-of-sample (or out of fold) scores, which are accessible through `fetch_oos_scores`. These may be stored in a Civis table on Redshift using the `oos_scores`, `oos_scores_db`, and `oos_scores_if_exists` parameters.

Predictions

A fitted model can be used to make predictions for data residing in any of the sources above and a `civis_file_manifest`. Similar to `civis_ml`, use the data source helpers as the `newdata` argument to `predict.civis_ml`.

A manifest file is a JSON file which specifies the location of many shards of the data to be used for prediction. A manifest file is the output of a Civis export job with `force_multifile = TRUE` set, e.g. from `civis_to_multifile.csv`. Large civis tables (provided using `table_name`) will automatically be exported to manifest files.

Prediction outputs will always be stored as gzipped CSVs in one or more civis files. Provide an `output_table` (and optionally an `output_db`, if it’s different from `database_name`) to copy these predictions into a table on Redshift.

See Also

`civis_file`, `civis_table`, and `civis_file_manifest` for specifying data sources. `get_metric` to access model validation metrics. `fetch_logs` for retrieving logs for a (failed) model build, `fetch_oos_scores` for retrieving the out of sample (fold) scores for each training observation, and `fetch_predictions` for retrieving the predictions from a prediction job.

Examples

```r
## Not run:
# From a data frame:
m <- civis_ml(df, model_type = "sparse_logistic",
 dependent_variable = "Species")

# From a table:
m <- civis_ml(civis_table("schema.table", "database_name"),
 model_type = "sparse_logistic", dependent_variable = "Species",
 oos_scores_table = "schema.scores_table",
 oos_scores_if_exists = "drop")

# From a local file:
m <- civis_ml("path/to/file.csv", model_type = "sparse_logistic",
 dependent_variable = "Species")

# From a Civis file:
file_id <- write_civis_file("path/to/file.csv", name = "file.csv")
m <- civis_ml(civis_file(file_id), model_type = "sparse_logistic",
 dependent_variable = "Species")

pred_job <- predict(m, newdata = df)
```
pred_job <- predict(m, civis_table("schema.table", "database_name"),
  output_table = "schema.scores_table")
pred_job <- predict(m, civis_file(file_id),
  output_table = "schema.scores_table")
m <- civis_ml_fetch_existing(model_id = m$job$id, m$run$id)
logs <- fetch_logs(m)
yhat <- fetch_oos_scores(m)
yhat <- fetch_predictions(pred_job)
## End(Not run)

CIVIS_ML_CLASSIFIERS  List of classification models.

Description
List of classification models.

Usage
CIVIS_ML_CLASSIFIERS

Format
An object of class character of length 6.

civis_ml_extra_trees_classifier
  CivisML Extra Trees Classifier

Description
CivisML Extra Trees Classifier

Usage
civis_ml_extra_trees_classifier(
  x,
  dependent_variable,
  primary_key = NULL,
  excluded_columns = NULL,
  n_estimators = 500,
  criterion = c("gini", "entropy"),
  max_depth = NULL,
  min_samples_split = 2,
min_samples_leaf = 1,
min_weight_fraction_leaf = 0,
max_features = "sqrt",
max_leaf_nodes = NULL,
min_impurity_split = 1e-07,
bootstrap = FALSE,
random_state = 42,
class_weight = NULL,
fit_params = NULL,
cross_validation_parameters = NULL,
calibration = NULL,
oos_scores_table = NULL,
oos_scores_db = NULL,
oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
model_name = NULL,
cpu_requested = NULL,
memory_requested = NULL,
disk_requested = NULL,
notifications = NULL,
polling_interval = NULL,
verbose = FALSE,
civisml_version = "prod"
)

Arguments

x
See the Data Sources section below.
dependent_variable
The dependent variable of the training dataset. For a multi-target problem, this should be a vector of column names of dependent variables. Nulls in a single dependent variable will automatically be dropped.

primary_key
Optional, the unique ID (primary key) of the training dataset. This will be used to index the out-of-sample scores. In predict.civis_ml, the primary_key of the training task is used by default primary_key = NA. Use primary_key = NULL to explicitly indicate the data have no primary_key.

excluded_columns
Optional, a vector of columns which will be considered ineligible to be independent variables.

n_estimators
The number of boosting stages to perform. Gradient boosting is fairly robust to over-fitting, so a large number usually results in better predictive performance.

criterion
The function to measure the quality of a split. Supported criteria are gini for the Gini impurity and entropy for the information gain.

max_depth
Maximum depth of the individual regression estimators. The maximum depth limits the number of nodes in the tree. Tune this parameter for best performance. The best value depends on the interaction of the input variables.

min_samples_split
The minimum number of samples required to split an internal node. If an integer, then consider min_samples_split as the minimum number. If a float,
then \( \text{min\_samples\_split} \) is a percentage and \( \lceil \text{min\_samples\_split} \times n\_samples \rceil \) are the minimum number of samples for each split.

**min\_samples\_leaf**

The minimum number of samples required to be in a leaf node. If an integer, then consider \( \text{min\_samples\_leaf} \) as the minimum number. If a float, the \( \text{min\_samples\_leaf} \) is a percentage and \( \lceil \text{min\_samples\_leaf} \times n\_samples \rceil \) are the minimum number of samples for each leaf node.

**min\_weight\_fraction\_leaf**

The minimum weighted fraction of the sum total of weights required to be at a leaf node.

**max\_features**

The number of features to consider when looking for the best split.

- **integer** consider \( \text{max\_features} \) at each split.
- **float** then \( \text{max\_features} \) is a percentage and \( \text{max\_features} \times n\_features \) are considered at each split.
- **auto** then \( \text{max\_features} = \sqrt{n\_features} \)
- **sqrt** then \( \text{max\_features} = \sqrt{n\_features} \)
- **log2** then \( \text{max\_features} = \log2(n\_features) \)
- **NULL** then \( \text{max\_features} = n\_features \)

**max\_leaf\_nodes**

Grow trees with \( \text{max\_leaf\_nodes} \) in best-first fashion. Best nodes are defined as relative reduction to impurity. If \( \text{max\_leaf\_nodes} = \text{NULL} \) then unlimited number of leaf nodes.

**min\_impurity\_split**

Threshold for early stopping in tree growth. A node will split if its impurity is above the threshold, otherwise it is a leaf.

**bootstrap**

Whether bootstrap samples are used when building trees.

**random\_state**

The seed of the random number generator.

**class\_weight**

A list with class\_label = value pairs, or balanced. When class\_weight = "balanced", the class weights will be inversely proportional to the class frequencies in the input data as:

\[
\frac{n\_samples}{n\_classes \times \text{table}(y)}
\]

Note, the class weights are multiplied with sample\_weight (passed via fit\_params) if sample\_weight is specified.

**fit\_params**

Optional, a mapping from parameter names in the model’s fit method to the column names which hold the data, e.g. list(sample\_weight = 'survey\_weight\_column').

**cross\_validation\_parameters**

Optional, parameter grid for learner parameters, e.g. list(n\__estimators = c(100,200,500),learning\_rate = c(0.01,0.1),max\_depth = c(2,3)) or "hyperband" for supported models.

**calibration**

Optional, if not NULL, calibrate output probabilities with the selected method, sigmoid, or isotonic. Valid only with classification models.

**oos\_scores\_table**

Optional, if provided, store out-of-sample predictions on training set data to this Redshift "schema.tablename".
civis_ml_extra_trees_classifier

- **oos_scores_db**: Optional, the name of the database where the oos_scores_table will be created. If not provided, this will default to database_name.

- **oos_scores_if_exists**: Optional, action to take if oos_scores_table already exists. One of "fail", "append", "drop", or "truncate". The default is "fail".

- **model_name**: Optional, the prefix of the Platform modeling jobs. It will have "Train" or "Predict" added to become the Script title.

- **cpu_requested**: Optional, the number of CPU shares requested in the Civis Platform for training jobs or prediction child jobs. 1024 shares = 1 CPU.

- **memory_requested**: Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.

- **disk_requested**: Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.

- **notifications**: Optional, model status notifications. See scripts_post_custom for further documentation about email and URL notification.

- **polling_interval**: Check for job completion every this number of seconds.

- **verbose**: Optional, If TRUE, supply debug outputs in Platform logs and make prediction child jobs visible.

- **civisml_version**: Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production.

### Value

A civis_ml object, a list containing the following elements:

- **job**: job metadata from scripts_get_custom.
- **run**: run metadata from scripts_get_custom_runs.
- **outputs**: CivisML metadata from scripts_list_custom_runs_outputs containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.
- **metrics**: Parsed CivisML output from metrics.json containing metadata from validation. A list containing the following elements:
  - run list, metadata about the run.
  - data list, metadata about the training data.
  - model list, the fitted scikit-learn model with CV results.
  - metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
  - warnings list.
  - data_platform list, training data location.
- **model_info**: Parsed CivisML output from model_info.json containing metadata from training. A list containing the following elements:
  - run list, metadata about the run.
• data list, metadata about the training data.
• model list, the fitted scikit-learn model.
• metrics empty list.
• warnings list.
• data_platform list, training data location.

Data Sources

For building models with `civis_ml`, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a `data.frame` resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling `civis_ml`:

- **data.frame** `civis_ml(x = df,...)`
- **local csv file** `civis_ml(x = "path/to/data.csv",...)`
- **file in Civis Platform** `civis_ml(x = civis_file(1234))`
- **table in Civis Platform** `civis_ml(x = civis_table(table_name = "schema.table",database_name = "database"))`

Examples

```r
## Not run:
df <- iris
names(df) <- gsub("\."", ",", names(df))

m <- civis_ml_extra_trees_classifier(df,
dependent_variable = "Species",
n_estimators = 100,
max_depth = 5,
max_features = NULL)
yhat <- fetch_oos_scores(m)

# Grid Search
cv_params <- list(
n_estimators = c(100, 200, 500),
max_depth = c(2, 3))

m <- civis_ml_extra_trees_classifier(df,
dependent_variable = "Species",
max_features = NULL,
cross_validation_parameters = cv_params)

pred_info <- predict(m, civis_table("schema.table", "my_database"),
output_table = "schema.scores_table")

## End(Not run)
```
civis_ml_extra_trees_regressor

CivisML Extra Trees Regressor

Description

CivisML Extra Trees Regressor

Usage

civis_ml_extra_trees_regressor(
  x,
  dependent_variable,
  primary_key = NULL,
  excluded_columns = NULL,
  n_estimators = 500,
  criterion = c("mse", "mae"),
  max_depth = NULL,
  min_samples_split = 2,
  min_samples_leaf = 1,
  min_weight_fraction_leaf = 0,
  max_features = "sqrt",
  max_leaf_nodes = NULL,
  min_impurity_split = 1e-07,
  bootstrap = FALSE,
  random_state = 42,
  fit_params = NULL,
  cross_validation_parameters = NULL,
  oos_scores_table = NULL,
  oos_scores_db = NULL,
  oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
  model_name = NULL,
  cpuRequested = NULL,
  memory_requested = NULL,
  disk_requested = NULL,
  notifications = NULL,
  polling_interval = NULL,
  verbose = FALSE,
  civisml_version = "prod"
)

Arguments

x
dependent_variable

See the Data Sources section below.

The dependent variable of the training dataset. For a multi-target problem, this should be a vector of column names of dependent variables. Nulls in a single dependent variable will automatically be dropped.
primary_key
Optional, the unique ID (primary key) of the training dataset. This will be used to index the out-of-sample scores. In `predict.civis_ml`, the `primary_key` of the training task is used by default `primary_key = NA`. Use `primary_key = NULL` to explicitly indicate the data have no primary_key.

excluded_columns
Optional, a vector of columns which will be considered ineligible to be independent variables.

n_estimators
The number of boosting stages to perform. Gradient boosting is fairly robust to over-fitting, so a large number usually results in better predictive performance.

criterion
The function used to measure the quality of a split. Supported criteria are `mse` for the mean squared error, and `mae` for the mean absolute error.

max_depth
Maximum depth of the individual regression estimators. The maximum depth limits the number of nodes in the tree. Tune this parameter for best performance. The best value depends on the interaction of the input variables.

min_samples_split
The minimum number of samples required to split an internal node. If an integer, then consider `min_samples_split` as the minimum number. If a float, then `min_samples_split` is a percentage and ceiling(`min_samples_split * n_samples`) are the minimum number of samples for each split.

min_samples_leaf
The minimum number of samples required to be in a leaf node. If an integer, then consider `min_samples_leaf` as the minimum number. If a float, the `min_samples_leaf` is a percentage and ceiling(`min_samples_leaf * n_samples`) are the minimum number of samples for each leaf node.

min_weight_fraction_leaf
The minimum weighted fraction of the sum total of weights required to be at a leaf node.

max_features
The number of features to consider when looking for the best split.

  integer consider `max_features` at each split.
  float then `max_features` is a percentage and `max_features * n_features` are considered at each split.
  auto then `max_features = sqrt(n_features)`
  sqrt then `max_features = sqrt(n_features)`
  log2 then `max_features = log2(n_features)`
  NULL then `max_features = n_features`

max_leaf_nodes
Grow trees with `max_leaf_nodes` in best-first fashion. Best nodes are defined as relative reduction to impurity. If `max_leaf_nodes = NULL` then unlimited number of leaf nodes.

min_impurity_split
Threshold for early stopping in tree growth. A node will split if its impurity is above the threshold, otherwise it is a leaf.

bootstrap
Whether bootstrap samples are used when building trees.

random_state
The seed of the random number generator.

fit_params
Optional, a mapping from parameter names in the model’s fit method to the column names which hold the data, e.g. `list(sample_weight = 'survey_weight_column').`
civis_ml_extra_trees_regressor

cross_validation_parameters
Optional, parameter grid for learner parameters, e.g. list(n_estimators = c(100,200,500),learning_rate = c(0.01,0.1),max_depth = c(2,3)) or "hyperband" for supported models.

oos_scores_table
Optional, if provided, store out-of-sample predictions on training set data to this Redshift "schema.tablename".

oos_scores_db
Optional, the name of the database where the oos_scores_table will be created. If not provided, this will default to database_name.

oos_scores_if_exists
Optional, action to take if oos_scores_table already exists. One of "fail", "append", "drop", or "truncate". The default is "fail".

model_name
Optional, the prefix of the Platform modeling jobs. It will have "Train" or "Predict" added to become the Script title.

cpu_requested
Optional, the number of CPU shares requested in the Civis Platform for training jobs or prediction child jobs. 1024 shares = 1 CPU.

memory_requested
Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.

disk_requested
Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.

notifications
Optional, model status notifications. See scripts_post_custom for further documentation about email and URL notification.

polling_interval
Check for job completion every this number of seconds.

verbose
Optional, If TRUE, supply debug outputs in Platform logs and make prediction child jobs visible.

civisml_version
Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production

Value
A civis_ml object, a list containing the following elements:

job
job metadata from scripts_get_custom.

run
run metadata from scripts_get_custom_runs.

outputs
CivisML metadata from scripts_list_custom_runs_outputs containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.

metrics
Parsed CivisML output from metrics.json containing metadata from validation. A list containing the following elements:
• run list, metadata about the run.
• data list, metadata about the training data.
• model list, the fitted scikit-learn model with CV results.
civis_ml_extra_trees_regressor

- metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
- warnings list.
- data_platform list, training data location.

**model_info**

Parsed CivisML output from model_info.json containing metadata from training. A list containing the following elements:

- run list, metadata about the run.
- data list, metadata about the training data.
- model list, the fitted scikit-learn model.
- metrics empty list.
- warnings list.
- data_platform list, training data location.

**Data Sources**

For building models with civis_ml, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a data.frame resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling civis_ml:

```r
data.frame civis_ml(x = df,...)
local csv file civis_ml(x = "path/to/data.csv",...)
file in Civis Platform civis_ml(x = civis_file(1234))
table in Civis Platform civis_ml(x = civis_table(table_name = "schema.table",database_name = "database"))
```

**Examples**

```r
## Not run:
data(ChickWeight)

m <- civis_ml_extra_trees_regressor(ChickWeight,
dependent_variable = "weight",
n_estimators = 100,
max_depth = 5,
max_features = NULL)
yhat <- fetch_oos_scores(m)

# Grid Search
cv_params <- list(
  n_estimators = c(100, 200, 500),
  max_depth = c(2, 3))

m <- civis_ml_extra_trees_regressor(ChickWeight,
dependent_variable = "weight",
max_features = NULL,
cross_validation_parameters = cv_params)

pred_info <- predict(m, civis_table("schema.table", "my_database"),
```
civis_ml_gradient_boosting_classifier

## Description
CivisML Gradient Boosting Classifier

## Usage

civis_ml_gradient_boosting_classifier(
    x,
    dependent_variable,
    primary_key = NULL,
    excluded_columns = NULL,
    loss = c("deviance", "exponential"),
    learning_rate = 0.1,
    n_estimators = 500,
    subsample = 1,
    criterion = c("friedman_mse", "mse", "mae"),
    min_samples_split = 2,
    min_samples_leaf = 1,
    min_weight_fraction_leaf = 0,
    max_depth = 2,
    min_impurity_split = 1e-07,
    random_state = 42,
    max_features = "sqrt",
    max_leaf_nodes = NULL,
    presort = c("auto", TRUE, FALSE),
    fit_params = NULL,
    cross_validation_parameters = NULL,
    calibration = NULL,
    oos_scores_table = NULL,
    oos_scores_db = NULL,
    oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
    model_name = NULL,
    cpuRequested = NULL,
    memoryRequested = NULL,
    diskRequested = NULL,
    notifications = NULL,
    polling_interval = NULL,
    verbose = FALSE,
    civisml_version = "prod"
)
Arguments

Arguments:

- **x**: See the Data Sources section below.
- **dependent_variable**: The dependent variable of the training dataset. For a multi-target problem, this should be a vector of column names of dependent variables. Nulls in a single dependent variable will automatically be dropped.
- **primary_key**: Optional, the unique ID (primary key) of the training dataset. This will be used to index the out-of-sample scores. In predict.civis_ml, the primary_key of the training task is used by default primary_key = NA. Use primary_key = NULL to explicitly indicate the data have no primary_key.
- **excluded_columns**: Optional, a vector of columns which will be considered ineligible to be independent variables.
- **loss**: The loss function to be optimized. deviance refers to deviance (logistic regression) for classification with probabilistic outputs. For exponential, gradient boosting recovers the AdaBoost algorithm.
- **learning_rate**: The learning rate shrinks the contribution of each tree by learning_rate. There is a trade-off between learning_rate and n_estimators.
- **n_estimators**: The number of boosting stages to perform. Gradient boosting is fairly robust to over-fitting, so a large number usually results in better predictive performance.
- **subsample**: The fraction of samples to be used for fitting individual base learners. If smaller than 1.0, this results in Stochastic Gradient Boosting. subsample interacts with the parameter n_estimators. Choosing subsample < 1.0 leads to a reduction of variance and an increase in bias.
- **criterion**: The function to measure the quality of a split. The default value criterion = "friedman_mse" is generally the best as it can provide a better approximation in some cases.
- **min_samples_split**: The minimum number of samples required to split an internal node. If an integer, then consider min_samples_split as the minimum number. If a float, then min_samples_split is a percentage and ceiling(min_samples_split * n_samples) are the minimum number of samples for each split.
- **min_samples_leaf**: The minimum number of samples required to be in a leaf node. If an integer, then consider min_samples_leaf as the minimum number. If a float, the min_samples_leaf is a percentage and ceiling(min_samples_leaf * n_samples) are the minimum number of samples for each leaf node.
- **min_weight_fraction_leaf**: The minimum weighted fraction of the sum total of weights required to be at a leaf node.
- **max_depth**: Maximum depth of the individual regression estimators. The maximum depth limits the number of nodes in the tree. Tune this parameter for best performance. The best value depends on the interaction of the input variables.
- **min_impurity_split**: Threshold for early stopping in tree growth. A node will split if its impurity is above the threshold, otherwise it is a leaf.
random_state  The seed of the random number generator.
max_features  The number of features to consider when looking for the best split.
   integer  consider max_features at each split.
   float   then max_features is a percentage and max_features * n_features are
           considered at each split.
   auto    then max_features = sqrt(n_features)
   sqrt    then max_features = sqrt(n_features)
   log2    then max_features = log2(n_features)
   NULL   then max_features = n_features
max_leaf_nodes  Grow trees with max_leaf_nodes in best-first fashion. Best nodes are defined as
                relative reduction to impurity. If max_leaf_nodes = NULL then unlimited num-
                ber of leaf nodes.
presort  Whether to presort the data to speed up the finding of best splits in fitting.
fit_params  Optional, a mapping from parameter names in the model's fit method to the
           column names which hold the data, e.g. list(sample_weight = 'survey_weight_column').
cross_validation_parameters  Optional, parameter grid for learner parameters, e.g. list(n_estimators =
                               c(100,200,500),learning_rate = c(0.01,0.1),max_depth = c(2,3)) or "hyperband"
                               for supported models.
calibration  Optional, if not NULL, calibrate output probabilities with the selected method,
             sigmoid or isotonic. Valid only with classification models.

oos_scores_table  Optional, if provided, store out-of-sample predictions on training set data to this
                  Redshift "schema.tablename".
oos_scores_db  Optional, the name of the database where the oos_scores_table will be cre-
                 ated. If not provided, this will default to database_name.
oos_scores_if_exists  Optional, action to take if oos_scores_table already exists. One of "fail",
                      "append", "drop", or "truncate". The default is "fail".
model_name  Optional, the prefix of the Platform modeling jobs. It will have "Train" or "
             Predict" added to become the Script title.
cpu_requested  Optional, the number of CPU shares requested in the Civis Platform for training
               jobs or prediction child jobs. 1024 shares = 1 CPU.
memory_requested  Optional, the memory requested from Civis Platform for training jobs or predic-
                   tion child jobs, in MiB.
disk_requested  Optional, the disk space requested on Civis Platform for training jobs or predic-
                  tion child jobs, in GB.
notifications  Optional, model status notifications. See scripts_post_custom for further
               documentation about email and URL notification.
polling_interval  Check for job completion every this number of seconds.
verbose  Optional, If TRUE, supply debug outputs in Platform logs and make prediction
         child jobs visible.
civisml_version

Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production

Value

A civis_ml object, a list containing the following elements:

- **job**
  job metadata from `scripts_get_custom`.

- **run**
  run metadata from `scripts_get_custom_runs`.

- **outputs**
  CivisML metadata from `scripts_list_custom_runs_outputs` containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.

- **metrics**
  Parsed CivisML output from `metrics.json` containing metadata from validation. A list containing the following elements:
  - run list, metadata about the run.
  - data list, metadata about the training data.
  - model list, the fitted scikit-learn model with CV results.
  - metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
  - warnings list.
  - data_platform list, training data location.

- **model_info**
  Parsed CivisML output from `model_info.json` containing metadata from training. A list containing the following elements:
  - run list, metadata about the run.
  - data list, metadata about the training data.
  - model list, the fitted scikit-learn model.
  - metrics empty list.
  - warnings list.
  - data_platform list, training data location.

Data Sources

For building models with `civis_ml`, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a `data.frame` resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling `civis_ml`:

- **data.frame** `civis_ml(x = df,...)`
- **local csv file** `civis_ml(x = "path/to/data.csv",...)`
- **file in Civis Platform** `civis_ml(x = civis_file(1234))`
- **table in Civis Platform** `civis_ml(x = civis_table(table_name = "schema.table",database_name = "database"))`
Examples

```r
## Not run:
df <- iris
names(df) <- gsub("\."", ",", names(df))

m <- civis_ml_gradient_boosting_classifier(df,
    dependent_variable = "Species",
    learning_rate = .01,
    n_estimators = 100,
    subsample = .5,
    max_depth = 5,
    max_features = NULL)
yhat <- fetch_oos_scores(m)

# Grid Search
cv_params <- list(
    n_estimators = c(100, 200, 500),
    learning_rate = c(.01, .1),
    max_depth = c(2, 3))

m <- civis_ml_gradient_boosting_classifier(df,
    dependent_variable = "Species",
    subsample = .5,
    max_features = NULL,
    cross_validation_parameters = cv_params)
pred_info <- predict(m, civis_table("schema.table", "my_database"),
    output_table = "schema.scores_table")

## End(Not run)
```

civis_ml_gradient_boosting_regressor

CivisML Gradient Boosting Regressor

Description

CivisML Gradient Boosting Regressor

Usage

```r
civis_ml_gradient_boosting_regressor(
    x,
    dependent_variable,
    primary_key = NULL,
    excluded_columns = NULL,
    loss = c("ls", "lad", "huber", "quantile"),
    learning_rate = 0.1,
    n_estimators = 500,
```
```r
subsample = 1,
criterion = c("friedman_mse", "mse", "mae"),
min_samples_split = 2,
min_samples_leaf = 1,
min_weight_fraction_leaf = 0,
max_depth = 2,
min_impurity_split = 1e-07,
random_state = 42,
max_features = "sqrt",
alpha = 0.9,
max_leaf_nodes = NULL,
presort = c("auto", TRUE, FALSE),
fit_params = NULL,
cross_validation_parameters = NULL,
oos_scores_table = NULL,
oos_scores_db = NULL,
oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
model_name = NULL,
cpu_requested = NULL,
memory_requested = NULL,
disk_requested = NULL,
notifications = NULL,
polling_interval = NULL,
verbose = FALSE,
civisml_version = "prod"
)
```

**Arguments**

- **x**
  - See the Data Sources section below.

- **dependent_variable**
  - The dependent variable of the training dataset. For a multi-target problem, this should be a vector of column names of dependent variables. Nulls in a single dependent variable will automatically be dropped.

- **primary_key**
  - Optional, the unique ID (primary key) of the training dataset. This will be used to index the out-of-sample scores. In `predict.civis_ml`, the primary_key of the training task is used by default `primary_key = NA`. Use `primary_key = NULL` to explicitly indicate the data have no primary_key.

- **excluded_columns**
  - Optional, a vector of columns which will be considered ineligible to be independent variables.

- **loss**
  - The loss function to be optimized. l1s refers to least squares regression. lad (least absolute deviation) is a highly robust loss function solely based on order information of the input variables. huber is a combination of the two. quantile allows quantile regression (use alpha to specify the quantile).

- **learning_rate**
  - The learning rate shrinks the contribution of each tree by `learning_rate`. There is a trade-off between `learning_rate` and `n_estimators`.
The number of boosting stages to perform. Gradient boosting is fairly robust to over-fitting, so a large number usually results in better predictive performance.

The fraction of samples to be used for fitting individual base learners. If smaller than 1.0, this results in Stochastic Gradient Boosting. subsample interacts with the parameter n_estimators. Choosing subsample < 1.0 leads to a reduction of variance and an increase in bias.

The function to measure the quality of a split. The default value criterion = "friedman_mse" is generally the best as it can provide a better approximation in some cases.

The minimum number of samples required to split an internal node. If an integer, then consider min_samples_split as the minimum number. If a float, then min_samples_split is a percentage and ceiling(min_samples_split * n_samples) are the minimum number of samples for each split.

The minimum number of samples required to be in a leaf node. If an integer, then consider min_samples_leaf as the minimum number. If a float, the min_samples_leaf is a percentage and ceiling(min_samples_leaf * n_samples) are the minimum number of samples for each leaf node.

The minimum weighted fraction of the sum total of weights required to be at a leaf node.

Maximum depth of the individual regression estimators. The maximum depth limits the number of nodes in the tree. Tune this parameter for best performance. The best value depends on the interaction of the input variables.

Threshold for early stopping in tree growth. A node will split if its impurity is above the threshold, otherwise it is a leaf.

The seed of the random number generator.

The number of features to consider when looking for the best split. integer consider max_features at each split.

float then max_features is a percentage and max_features * n_features are considered at each split.

auto then max_features = sqrt(n_features)

sqrt then max_features = sqrt(n_features)

log2 then max_features = log2(n_features)

NULL then max_features = n_features

The alpha-quantile of the huber loss function and the quantile loss function. Ignored unless loss = "huber" or loss = "quantile"

Grow trees with max_leaf_nodes in best-first fashion. Best nodes are defined as relative reduction to impurity. If max_leaf_nodes = NULL then unlimited number of leaf nodes.

Whether to presort the data to speed up the finding of best splits in fitting.

Optional, a mapping from parameter names in the model's fit method to the column names which hold the data, e.g. list(sample_weight = 'survey_weight_column').
cross_validation_parameters

Optional, parameter grid for learner parameters, e.g., `list(n_estimators = c(100,200,500), learning_rate = c(0.01,0.1), max_depth = c(2,3))` or "hyperband" for supported models.

**oos_scores_table**

Optional, if provided, store out-of-sample predictions on training set data to this Redshift "schema.tablename".

**oos_scores_db**

Optional, the name of the database where the oos_scores_table will be created. If not provided, this will default to database_name.

**oos_scores_if_exists**

Optional, action to take if oos_scores_table already exists. One of "fail", "append", "drop", or "truncate". The default is "fail".

**model_name**

Optional, the prefix of the Platform modeling jobs. It will have "Train" or "Predict" added to become the Script title.

**cpu_requested**

Optional, the number of CPU shares requested in the Civis Platform for training jobs or prediction child jobs. 1024 shares = 1 CPU.

**memory_requested**

Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.

**disk_requested**

Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.

**notifications**

Optional, model status notifications. See scripts_post_custom for further documentation about email and URL notification.

**polling_interval**

Check for job completion every this number of seconds.

**verbose**

Optional, If TRUE, supply debug outputs in Platform logs and make prediction child jobs visible.

**civisml_version**

Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production

**Value**

A civis_ml object, a list containing the following elements:

- **job**
  job metadata from scripts_get_custom.

- **run**
  run metadata from scripts_get_custom_runs.

- **outputs**
  CivisML metadata from scripts_list_custom_runs_outputs containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.

- **metrics**
  Parsed CivisML output from metrics.json containing metadata from validation. A list containing the following elements:
  - run list, metadata about the run.
  - data list, metadata about the training data.
  - model list, the fitted scikit-learn model with CV results.
civis_ml_gradient_boostingRegressor

• metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
• warnings list.
• data_platform list, training data location.

model_info

Parsed CivisML output from model_info.json containing metadata from training. A list containing the following elements:
• run list, metadata about the run.
• data list, metadata about the training data.
• model list, the fitted scikit-learn model.
• metrics empty list.
• warnings list.
• data_platform list, training data location.

Data Sources

For building models with civis_ml, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a data.frame resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling civis_ml:

data.frame  civis_ml(x = df,...)
local csv file  civis_ml(x = "path/to/data.csv",...)
file in Civis Platform  civis_ml(x = civis_file(1234))
table in Civis Platform  civis_ml(x = civis_table(table_name = "schema.table",database_name = "database"))

Examples

## Not run:
data(ChickWeight)
m <- civis_ml_gradient_boostingRegressor(ChickWeight,
dependent_variable = "weight",
learning_rate = .01,
n_estimators = 100,
subsample = .5,
max_depth = 5,
max_features = NULL)
yhat <- fetch_oos_scores(m)

# Grid Search
cv_params <- list(
  n_estimators = c(100, 200, 500),
  learning_rate = c(.01, .1),
  max_depth = c(2, 3))
m <- civis_ml_gradient_boostingRegressor(ChickWeight,
dependent_variable = "weight",
subsample = .5,
max_features = NULL,
cross_validation_parameters = cv_params)
pred_info <- predict(m, civis_table("schema.table", "my_database"),
output_table = "schema.scores_table")

## End(Not run)

---

civis_ml_random_forest_classifier

*CivisML Random Forest Classifier*

**Description**

CivisML Random Forest Classifier

**Usage**

civis_ml_random_forest_classifier(
  x,
  dependent_variable,
  primary_key = NULL,
  excluded_columns = NULL,
  n_estimators = 500,
  criterion = c("gini", "entropy"),
  max_depth = NULL,
  min_samples_split = 2,
  min_samples_leaf = 1,
  min_weight_fraction_leaf = 0,
  max_features = "sqrt",
  max_leaf_nodes = NULL,
  min_impurity_split = 1e-07,
  bootstrap = TRUE,
  random_state = 42,
  class_weight = NULL,
  fit_params = NULL,
  cross_validation_parameters = NULL,
  calibration = NULL,
  oos_scores_table = NULL,
  oos_scores_db = NULL,
  oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
  model_name = NULL,
  cpu_requested = NULL,
  memory_requested = NULL,
  disk_requested = NULL,
  notifications = NULL,
  polling_interval = NULL,
civis_ml_random_forest_classifier

verbose = FALSE,
civisml_version = "prod"
)

Arguments

x
dependent_variable
The dependent variable of the training dataset. For a multi-target problem, this should be a vector of column names of dependent variables. Nulls in a single dependent variable will automatically be dropped.

primary_key
Optional, the unique ID (primary key) of the training dataset. This will be used to index the out-of-sample scores. In predict.civis_ml, the primary_key of the training task is used by default primary_key = NA. Use primary_key = NULL to explicitly indicate the data have no primary_key.

excluded_columns
Optional, a vector of columns which will be considered ineligible to be independent variables.

n_estimators
The number of boosting stages to perform. Gradient boosting is fairly robust to over-fitting, so a large number usually results in better predictive performance.

criterion
The function to measure the quality of a split. Supported criteria are gini for the Gini impurity and entropy for the information gain.

max_depth
Maximum depth of the individual regression estimators. The maximum depth limits the number of nodes in the tree. Tune this parameter for best performance. The best value depends on the interaction of the input variables.

min_samples_split
The minimum number of samples required to split an internal node. If an integer, then consider min_samples_split as the minimum number. If a float, then min_samples_split is a percentage and ceiling(min_samples_split * n_samples) are the minimum number of samples for each split.

min_samples_leaf
The minimum number of samples required to be in a leaf node. If an integer, then consider min_samples_leaf as the minimum number. If a float, the min_samples_leaf is a percentage and ceiling(min_samples_leaf * n_samples) are the minimum number of samples for each leaf node.

min_weight_fraction_leaf
The minimum weighted fraction of the sum total of weights required to be at a leaf node.

max_features
The number of features to consider when looking for the best split.

integer consider max_features at each split.
float then max_features is a percentage and max_features * n_features are considered at each split.
auto then max_features = sqrt(n_features)
sqrt then max_features = sqrt(n_features)
log2 then max_features = log2(n_features)
**null** then `max_features = n_features`

`max_leaf_nodes` Grow trees with `max_leaf_nodes` in best-first fashion. Best nodes are defined as relative reduction to impurity. If `max_leaf_nodes = NULL` then unlimited number of leaf nodes.

`min_impurity_split` Threshold for early stopping in tree growth. A node will split if its impurity is above the threshold, otherwise it is a leaf.

`bootstrap` Whether bootstrap samples are used when building trees.

`random_state` The seed of the random number generator.

`class_weight` A list with `class_label = value` pairs, or balanced. When `class_weight = "balanced"`, the class weights will be inversely proportional to the class frequencies in the input data as:

\[
\frac{n_{samples}}{n_{classes} * \text{table}(y)}
\]

Note, the class weights are multiplied with `sample_weight` (passed via `fit_params`) if `sample_weight` is specified.

`fit_params` Optional, a mapping from parameter names in the model’s `fit` method to the column names which hold the data, e.g. `list(sample_weight = 'survey_weight_column')`.

`cross_validation_parameters` Optional, parameter grid for learner parameters, e.g. `list(n_estimators = c(100,200,500),learning_rate = c(0.01,0.1),max_depth = c(2,3))` or "hyperband" for supported models.

`calibration` Optional, if not `NULL`, calibrate output probabilities with the selected method, sigmoid, or isotonic. Valid only with classification models.

`oos_scores_table` Optional, if provided, store out-of-sample predictions on training set data to this Redshift "schema.tablename".

`oos_scores_db` Optional, the name of the database where the `oos_scores_table` will be created. If not provided, this will default to `database_name`.

`oos_scores_if_exists` Optional, action to take if `oos_scores_table` already exists. One of "fail", "append", "drop", or "truncate". The default is "fail".

`model_name` Optional, the prefix of the Platform modeling jobs. It will have "Train" or "Predict" added to become the Script title.

`cpu_requested` Optional, the number of CPU shares requested in the Civis Platform for training jobs or prediction child jobs. 1024 shares = 1 CPU.

`memory_requested` Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.

`disk_requested` Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.

`notifications` Optional, model status notifications. See `scripts_post_custom` for further documentation about email and URL notification.
polling_interval  Check for job completion every this number of seconds.
verbose  Optional. If TRUE, supply debug outputs in Platform logs and make prediction child jobs visible.
civisml_version  Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production

Value
A civis_ml object, a list containing the following elements:

job  job metadata from scripts_get_custom.
run  run metadata from scripts_get_custom_runs.
outputs  CivisML metadata from scripts_list_custom_runs_outputs containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.
metrics  Parsed CivisML output from metrics.json containing metadata from validation. A list containing the following elements:
  • run list, metadata about the run.
  • data list, metadata about the training data.
  • model list, the fitted scikit-learn model with CV results.
  • metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
  • warnings list.
  • data_platform list, training data location.
model_info  Parsed CivisML output from model_info.json containing metadata from training. A list containing the following elements:
  • run list, metadata about the run.
  • data list, metadata about the training data.
  • model list, the fitted scikit-learn model.
  • metrics empty list.
  • warnings list.
  • data_platform list, training data location.

Data Sources
For building models with civis_ml, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a data.frame resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling civis_ml:

data.frame  civis_ml(x = df,...)
local csv file  civis_ml(x = "path/to/data.csv",...)
file in Civis Platform  civis_ml(x = civis_file(1234))
table in Civis Platform  civis_ml(x = civis_table(table_name = "schema.table",database_name = "database"))
Examples

```r
## Not run:
df <- iris
names(df) <- gsub("\."", ",", names(df))

m <- civis_ml_random_forest_classifier(df,
  dependent_variable = "Species",
  n_estimators = 100,
  max_depth = 5,
  max_features = NULL)
yhat <- fetch_oos_scores(m)

# Grid Search
cv_params <- list(
  n_estimators = c(100, 200, 500),
  max_depth = c(2, 3))

m <- civis_ml_random_forest_classifier(df,
  dependent_variable = "Species",
  max_features = NULL,
  cross_validation_parameters = cv_params)
pred_info <- predict(m, civis_table("schema.table", "my_database"),
  output_table = "schema.scores_table")

## End(Not run)
```

civis_ml_random_forest_regressor

*CivisML Random Forest Regressor*

Description

CivisML Random Forest Regressor

Usage

```r
civis_ml_random_forest_regressor(
  x,
  dependent_variable,
  primary_key = NULL,
  excluded_columns = NULL,
  n_estimators = 500,
  criterion = c("mse", "mae"),
  max_depth = NULL,
  min_samples_split = 2,
  min_samples_leaf = 1,
  min_weight_fraction_leaf = 0,
)```
max_features = "sqrt",
max_leaf_nodes = NULL,
min_impurity_split = 1e-07,
bootstrap = TRUE,
random_state = 42,
fit_params = NULL,
cross_validation_parameters = NULL,
oos_scores_table = NULL,
oos_scores_db = NULL,
oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
model_name = NULL,
cpu_requested = NULL,
memory_requested = NULL,
disk_requested = NULL,
notifications = NULL,
polling_interval = NULL,
verbose = FALSE,
civisml_version = "prod"
)

Arguments

x See the Data Sources section below.
dependent_variable
The dependent variable of the training dataset. For a multi-target problem, this
should be a vector of column names of dependent variables. Nulls in a single
dependent variable will automatically be dropped.

primary_key Optional, the unique ID (primary key) of the training dataset. This will be used
to index the out-of-sample scores. In predict.civis_ml, the primary_key of
the training task is used by default primary_key = NA. Use primary_key = NULL
to explicitly indicate the data have no primary_key.

excluded_columns Optional, a vector of columns which will be considered ineligible to be indepen-
dent variables.
n_estimators The number of boosting stages to perform. Gradient boosting is fairly robust to
over-fitting, so a large number usually results in better predictive performance.
criterion The function used to measure the quality of a split. Supported criteria are mse
for the mean squared error, and mae for the mean absolute error.
max_depth Maximum depth of the individual regression estimators. The maximum depth
limits the number of nodes in the tree. Tune this parameter for best performance.
The best value depends on the interaction of the input variables.
min_samples_split The minimum number of samples required to split an internal node. If an in-
teger, then consider min_samples_split as the minimum number. If a float,
then min_samples_split is a percentage and ceiling(min_samples_split *
n_samples) are the minimum number of samples for each split.
The minimum number of samples required to be in a leaf node. If an integer, then consider `min_samples_leaf` as the minimum number. If a float, the `min_samples_leaf` is a percentage and `ceiling(min_samples_leaf * n_samples)` are the minimum number of samples for each leaf node.

`min_weight_fraction_leaf`

The minimum weighted fraction of the sum total of weights required to be at a leaf node.

`max_features`

The number of features to consider when looking for the best split.

- integer: consider `max_features` at each split.
- float: then `max_features` is a percentage and `max_features * n_features` are considered at each split.
- `auto`: then `max_features = sqrt(n_features)`
- `sqrt`: then `max_features = sqrt(n_features)`
- `log2`: then `max_features = log2(n_features)`
- `NULL`: then `max_features = n_features`

`max_leaf_nodes`

Grow trees with `max_leaf_nodes` in best-first fashion. Best nodes are defined as relative reduction to impurity. If `max_leaf_nodes = NULL` then unlimited number of leaf nodes.

`min_impurity_split`

Threshold for early stopping in tree growth. A node will split if its impurity is above the threshold, otherwise it is a leaf.

`bootstrap`

Whether bootstrap samples are used when building trees.

`random_state`

The seed of the random number generator.

`fit_params`

Optional, a mapping from parameter names in the model's `fit` method to the column names which hold the data, e.g. `list(sample_weight = 'survey_weight_column')`.

`cross_validation_parameters`

Optional, parameter grid for learner parameters, e.g. `list(n_estimators = c(100, 200, 500), learning_rate = c(0.01, 0.1), max_depth = c(2, 3))` or "hyperband" for supported models.

`oos_scores_table`

Optional, if provided, store out-of-sample predictions on training set data to this Redshift "schema.tablename".

`oos_scores_db`

Optional, the name of the database where the `oos_scores_table` will be created. If not provided, this will default to `database_name`.

`oos_scores_if_exists`

Optional, action to take if `oos_scores_table` already exists. One of "fail", "append", "drop", or "truncate". The default is "fail".

`model_name`

Optional, the prefix of the Platform modeling jobs. It will have "Train" or "Predict" added to become the Script title.

`cpu_requested`

Optional, the number of CPU shares requested in the Civis Platform for training jobs or prediction child jobs. 1024 shares = 1 CPU.

`memory_requested`

Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.
**disk_requested**  
Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.

**notifications**  
Optional, model status notifications. See `scripts_post_custom` for further documentation about email and URL notification.

**polling_interval**  
Check for job completion every this number of seconds.

**verbose**  
Optional, If `TRUE`, supply debug outputs in Platform logs and make prediction child jobs visible.

**civisml_version**  
Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production

**Value**

A `civis_ml` object, a list containing the following elements:

- **job**  
  job metadata from `scripts_get_custom`.

- **run**  
  run metadata from `scripts_get_custom_runs`.

- **outputs**  
  CivisML metadata from `scripts_list_custom_runs_outputs` containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.

- **metrics**  
  Parsed CivisML output from `metrics.json` containing metadata from validation. A list containing the following elements:
  - run list, metadata about the run.
  - data list, metadata about the training data.
  - model list, the fitted scikit-learn model with CV results.
  - metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
  - warnings list.
  - data_platform list, training data location.

- **model_info**  
  Parsed CivisML output from `model_info.json` containing metadata from training. A list containing the following elements:
  - run list, metadata about the run.
  - data list, metadata about the training data.
  - model list, the fitted scikit-learn model.
  - metrics empty list.
  - warnings list.
  - data_platform list, training data location.

**Data Sources**

For building models with `civis_ml`, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a `data.frame` resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling `civis_ml`:
data.frame civis_ml(x = df, ...)

local csv file civis_ml(x = "path/to/data.csv", ...)  

file in Civis Platform civis_ml(x = civis_file(1234))

table in Civis Platform civis_ml(x = civis_table(table_name = "schema.table", database_name = "database"))

Examples

```r
## Not run:

data(ChickWeight)

m <- civis_ml_random_forest_regressor(ChickWeight,  
  dependent_variable = "weight",  
  n_estimators = 100,  
  max_depth = 5,  
  max_features = NULL)

yhat <- fetch_oos_scores(m)

# Grid Search
cv_params <- list(  
  n_estimators = c(100, 200, 500),  
  max_depth = c(2, 3))

m <- civis_ml_random_forest_regressor(ChickWeight,  
  dependent_variable = "weight",  
  max_features = NULL,  
  cross_validation_parameters = cv_params)

pred_info <- predict(m, civis_table("schema.table", "my_database"),  
  output_table = "schema.scores_table")

## End(Not run)
```

data.frame civis_ml(x = df, ...)
local csv file civis_ml(x = "path/to/data.csv", ...)
file in Civis Platform civis_ml(x = civis_file(1234))
table in Civis Platform civis_ml(x = civis_table(table_name = "schema.table", database_name = "database"))

Examples

```r
## Not run:

data(ChickWeight)

m <- civis_ml_random_forest_regressor(ChickWeight,  
  dependent_variable = "weight",  
  n_estimators = 100,  
  max_depth = 5,  
  max_features = NULL)

yhat <- fetch_oos_scores(m)

# Grid Search
cv_params <- list(  
  n_estimators = c(100, 200, 500),  
  max_depth = c(2, 3))

m <- civis_ml_random_forest_regressor(ChickWeight,  
  dependent_variable = "weight",  
  max_features = NULL,  
  cross_validation_parameters = cv_params)

pred_info <- predict(m, civis_table("schema.table", "my_database"),  
  output_table = "schema.scores_table")

## End(Not run)
```
CivisML Sparse Linear Regression

Usage

```r
civis_ml_sparse_linear_regressor(
  x,  
  dependent_variable,  
  primary_key = NULL,  
  excluded_columns = NULL,  
  fit_intercept = TRUE,  
  normalize = FALSE,  
  fit_params = NULL,  
  cross_validation_parameters = NULL,  
  oos_scores_table = NULL,  
  oos_scores_db = NULL,  
  oos_scores_if_exists = c("fail", "append", "drop", "truncate"),  
  model_name = NULL,  
  cpu_requested = NULL,  
  memory_requested = NULL,  
  disk_requested = NULL,  
  notifications = NULL,  
  polling_interval = NULL,  
  verbose = FALSE,  
  civisml_version = "prod"
)
```

Arguments

- `x`  
  See the Data Sources section below.

- `dependent_variable`  
  The dependent variable of the training dataset. For a multi-target problem, this should be a vector of column names of dependent variables. Nulls in a single dependent variable will automatically be dropped.

- `primary_key`  
  Optional, the unique ID (primary key) of the training dataset. This will be used to index the out-of-sample scores. In `predict.civis_ml`, the `primary_key` of the training task is used by default `primary_key = NA`. Use `primary_key = NULL` to explicitly indicate the data have no `primary_key`.

- `excluded_columns`  
  Optional, a vector of columns which will be considered ineligible to be independent variables.
**civis_ml_sparse_linear_regressor**

- **fit_intercept**: Should an intercept term be included in the model? If FALSE, no intercept will be included, in this case the data are expected to already be centered.
- **normalize**: If TRUE, the regressors will be normalized before fitting the model. normalize is ignored when fit_intercept = FALSE.
- **fit_params**: Optional, a mapping from parameter names in the model’s fit method to the column names which hold the data, e.g. list(sample_weight = 'survey_weight_column').
- **cross_validation_parameters**: Optional, parameter grid for learner parameters, e.g. list(n_estimators = c(100,200,500), learning_rate = c(0.01,0.1), max_depth = c(2,3)) or "hyperband" for supported models.
- **oos_scores_table**: Optional, if provided, store out-of-sample predictions on training set data to this Redshift "schema.tablename".
- **oos_scores_db**: Optional, the name of the database where the oos_scores_table will be created. If not provided, this will default to database_name.
- **oos_scores_if_exists**: Optional, action to take if oos_scores_table already exists. One of "fail", "append", "drop", or "truncate". The default is "fail".
- **model_name**: Optional, the prefix of the Platform modeling jobs. It will have "Train" or "Predict" added to become the Script title.
- **cpu_requested**: Optional, the number of CPU shares requested in the Civis Platform for training jobs or prediction child jobs. 1024 shares = 1 CPU.
- **memory_requested**: Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.
- **disk_requested**: Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.
- **notifications**: Optional, model status notifications. See scripts_post_custom for further documentation about email and URL notification.
- **polling_interval**: Check for job completion every this number of seconds.
- **verbose**: Optional, If TRUE, supply debug outputs in Platform logs and make prediction child jobs visible.
- **civisml_version**: Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production.

**Value**

A civis_ml object, a list containing the following elements:

- **job**: job metadata from scripts_get_custom.
- **run**: run metadata from scripts_get_custom.runs.
- **outputs**: CivisML metadata from scripts_list_custom_runs_outputs containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.
metrics

Parsed CivisML output from metrics.json containing metadata from validation. A list containing the following elements:

- run list, metadata about the run.
- data list, metadata about the training data.
- model list, the fitted scikit-learn model with CV results.
- metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
- warnings list.
- data_platform list, training data location.

model_info

Parsed CivisML output from model_info.json containing metadata from training. A list containing the following elements:

- run list, metadata about the run.
- data list, metadata about the training data.
- model list, the fitted scikit-learn model.
- metrics empty list.
- warnings list.
- data_platform list, training data location.

Data Sources

For building models with civis_ml, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a data.frame resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling civis_ml:

data.frame civis_ml(x = df,...)

local csv file civis_ml(x = "path/to/data.csv",...)

file in Civis Platform civis_ml(x = civis_file(1234))

table in Civis Platform civis_ml(x = civis_table(table_name = "schema.table",database_name = "database"))

Examples

```r
## Not run:
data(ChickWeight)
m <- civis_ml_sparse_linear_regressor(ChickWeight, dependent_variable = "weight")
yhat <- fetch_oos_scores(m)

# make a prediction job, storing in a redshift table
pred_info <- predict(m, newdata = civis_table("schema.table", "my_database"),
                      output_table = "schema.scores_table")
```

## End(Not run)
civis_ml_sparse_logistic

CivisML Sparse Logistic

Description

CivisML Sparse Logistic

Usage

civis_ml_sparse_logistic(
  x,
  dependent_variable,
  primary_key = NULL,
  excluded_columns = NULL,
  penalty = c("l2", "l1"),
  dual = FALSE,
  tol = 1e-08,
  C = 499999950,
  fit_intercept = TRUE,
  intercept_scaling = 1,
  class_weight = NULL,
  random_state = 42,
  solver = c("liblinear", "newton-cg", "lbfgs", "sag"),
  max_iter = 100,
  multi_class = c("ovr", "multinomial"),
  fit_params = NULL,
  cross_validation_parameters = NULL,
  calibration = NULL,
  oos_scores_table = NULL,
  oos_scores_db = NULL,
  oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
  model_name = NULL,
  cpu_requested = NULL,
  memory_requested = NULL,
  disk_requested = NULL,
  notifications = NULL,
  polling_interval = NULL,
  verbose = FALSE,
  civisml_version = "prod"
)

Arguments

x See the Data Sources section below.
dependent_variable
The dependent variable of the training dataset. For a multi-target problem, this should be a vector of column names of dependent variables. Nulls in a single dependent variable will automatically be dropped.

primary_key
Optional, the unique ID (primary key) of the training dataset. This will be used to index the out-of-sample scores. In predict.civis_ml, the primary_key of the training task is used by default primary_key = NA. Use primary_key = NULL to explicitly indicate the data have no primary_key.

excluded_columns
Optional, a vector of columns which will be considered ineligible to be independent variables.

penalty
Used to specify the norm used in the penalization. The newton-cg, sag, and lbfgs solvers support only l2 penalties.

dual
Dual or primal formulation. Dual formulation is only implemented for l2 penalty with the liblinear solver. dual = FALSE should be preferred when n_samples > n_features.

tol
Tolerance for stopping criteria.

C
Inverse of regularization strength, must be a positive float. Smaller values specify stronger regularization.

fit_intercept
Should a constant or intercept term be included in the model.

intercept_scaling
Useful only when the solver = "liblinear" and fit_intercept = TRUE. In this case, a constant term with the value intercept_scaling is added to the design matrix.

class_weight
A list with class_label = value pairs, or balanced. When class_weight = "balanced", the class weights will be inversely proportional to the class frequencies in the input data as:

\[
\frac{n_{samples}}{n_{classes} \ast \text{table}(y)}
\]

Note, the class weights are multiplied with sample_weight (passed via fit_params) if sample_weight is specified.

random_state
The seed of the random number generator to use when shuffling the data. Used only in solver = "sag" and solver = "liblinear".

solver
Algorithm to use in the optimization problem. For small data liblinear is a good choice. sag is faster for larger problems. For multiclass problems, only newton-cg, sag, and lbfgs handle multinomial loss. liblinear is limited to one-versus-rest schemes. newton-cg, lbfgs, and sag only handle the l2 penalty.

Note that sag fast convergence is only guaranteed on features with approximately the same scale.

max_iter
The maximum number of iterations taken for the solvers to converge. Useful for the newton-cg, sag, and lbfgs solvers.
multi_class

The scheme for multi-class problems. When ovr, then a binary problem is fit for each label. When multinomial, a single model is fit minimizing the multinomial loss. Note, multinomial only works with the newton-cg, sag, and lbfgs solvers.

fit_params

Optional, a mapping from parameter names in the model’s fit method to the column names which hold the data, e.g. `list(sample_weight = 'survey_weight_column')`.

cross_validation_parameters

Optional, parameter grid for learner parameters, e.g. `list(n_estimators = c(100,200,500),learning_rate = c(0.01,0.1),max_depth = c(2,3))` or "hyperband" for supported models.

calibration

Optional, if not NULL, calibrate output probabilities with the selected method, sigmoid, or isotonic. Valid only with classification models.

oos_scores_table

Optional, if provided, store out-of-sample predictions on training set data to this Redshift "schema.tablename".

oos_scores_db

Optional, the name of the database where the oos_scores_table will be created. If not provided, this will default to database_name.

oos_scores_if_exists

Optional, action to take if oos_scores_table already exists. One of "fail", "append", "drop", or "truncate". The default is "fail".

model_name

Optional, the prefix of the Platform modeling jobs. It will have "Train" or "Predict" added to become the Script title.

cpu_requested

Optional, the number of CPU shares requested in the Civis Platform for training jobs or prediction child jobs. 1024 shares = 1 CPU.

memory_requested

Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.

disk_requested

Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.

notifications

Optional, model status notifications. See `scripts_post_custom` for further documentation about email and URL notification.

polling_interval

Check for job completion every this number of seconds.

verbose

Optional. If TRUE, supply debug outputs in Platform logs and make prediction child jobs visible.

civisml_version

Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production

Value

A `civis_ml` object, a list containing the following elements:

- `job` : job metadata from `scripts_get_custom`.
- `run` : run metadata from `scripts_get_custom_runs`.
civis_ml_sparse_logistic

outputs  
CivisML metadata from `scripts_list_custom_runs_outputs` containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.

metrics  
Parsed CivisML output from `metrics.json` containing metadata from validation. A list containing the following elements:
- run list, metadata about the run.
- data list, metadata about the training data.
- model list, the fitted scikit-learn model with CV results.
- metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
- warnings list.
- data_platform list, training data location.

model_info  
Parsed CivisML output from `model_info.json` containing metadata from training. A list containing the following elements:
- run list, metadata about the run.
- data list, metadata about the training data.
- model list, the fitted scikit-learn model.
- metrics empty list.
- warnings list.
- data_platform list, training data location.

Data Sources

For building models with `civis_ml`, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a `data.frame` resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling `civis_ml`:

- `data.frame`  `civis_ml(x = df,...)`
- `local csv file`  `civis_ml(x = "path/to/data.csv",...)`
- `file in Civis Platform`  `civis_ml(x = civis_file(1234))`
- `table in Civis Platform`  `civis_ml(x = civis_table(table_name = "schema.table",database_name = "database"))`

Examples

```r
## Not run:
df <- iris
names(df) <- gsub("\.", ",", names(df))

m <- civis_ml_sparse_logistic(df, "Species")
yhat <- fetch_oos_scores(m)

# Grid Search
cv_params <- list(C = c(.01, 1, 10, 100, 1000))
```

m <- civis_ml_sparse_logistic(df, "Species",
  cross_validation_parameters = cv_params)

# make a prediction job, storing in a redshift table
pred_info <- predict(m, newdata = civis_table("schema.table", "my_database"),
  output_table = "schema.scores_table")

## End(Not run)

civis_ml_sparse_ridge_regressor

*CivisML Sparse Ridge Regression*

**Description**

CivisML Sparse Ridge Regression

**Usage**

```r
civis_ml_sparse_ridge_regressor(
  x,
  dependent_variable,
  primary_key = NULL,
  excluded_columns = NULL,
  alpha = 1,
  fit_intercept = TRUE,
  normalize = FALSE,
  max_iter = NULL,
  tol = 0.001,
  solver = c("auto", "svd", "cholesky", "lsqr", "sparse_cg", "sag"),
  random_state = 42,
  fit_params = NULL,
  cross_validation_parameters = NULL,
  oos_scores_table = NULL,
  oos_scores_db = NULL,
  oos_scores_if_exists = c("fail", "append", "drop", "truncate"),
  model_name = NULL,
  cpu-requested = NULL,
  memory-requested = NULL,
  disk-requested = NULL,
  notifications = NULL,
  polling_interval = NULL,
  verbose = FALSE,
  civisml_version = "prod"
)
```
Arguments

x
  See the Data Sources section below.

dependent_variable
  The dependent variable of the training dataset. For a multi-target problem, this
  should be a vector of column names of dependent variables. Nulls in a single
  dependent variable will automatically be dropped.

primary_key
  Optional, the unique ID (primary key) of the training dataset. This will be used
  to index the out-of-sample scores. In predict.civis_ml, the primary_key of
  the training task is used by default primary_key = NA. Use primary_key = NULL
  to explicitly indicate the data have no primary_key.

excluded_columns
  Optional, a vector of columns which will be considered ineligible to be indepen-
  dent variables.

alpha
  The regularization strength, must be a vector of floats of length n_targets or a
  single float. Larger values specify stronger regularization.

fit_intercept
  Should an intercept term be included in the model. If FALSE, no intercept will
  be included, in this case the data are expected to already be centered.

normalize
  If TRUE, the regressors will be normalized before fitting the model. normalize
  is ignored when fit_intercept = FALSE.

max_iter
  Maximum number of iterations for conjugate gradient solver. For sparse_cg
  and lsqr solvers, the default value is predetermined. For the sag solver, the
  default value is 1000.

tol
  Precision of the solution.

solver
  Solver to use for the optimization problem.
  auto chooses the solver automatically based on the type of data.
  svd uses Singular Value Decomposition of X to compute the Ridge coefficients.
  More stable for singular matrices than cholesky.
  cholesky uses the standard decomposition to obtain a closed-form solution.
  sparse_cg uses the conjugate gradient solver. As an iterative algorithm, this
  solver is more appropriate than cholesky for large-scale data.
  lsqr uses the dedicated regularized least-squares routine.
  sag uses Stochastic Average Gradient descent. It also uses an iterative pro-
  cedure, and is often faster than other solvers when both n_samples and
  n_features are large. Note that sag fast convergence is only guaranteed on
  features with approximately the same scale

random_state
  The seed of the pseudo random number generator to use when shuffling the data.
  Used only when solver = “sag”.

fit_params
  Optional, a mapping from parameter names in the model’s fit method to the
  column names which hold the data, e.g. list(sample_weight = 'survey_weight_column').

cross_validation_parameters
  Optional, parameter grid for learner parameters, e.g. list(n_estimators =
  c(100,200,500),learning_rate = c(0.01,0.1),max_depth = c(2,3)) or "hyperband"
  for supported models.
oos_scores_table
Optional, if provided, store out-of-sample predictions on training set data to this Redshift "schema.tablename".

oos_scores_db
Optional, the name of the database where the oos_scores_table will be created. If not provided, this will default to database_name.

oos_scores_if_exists
Optional, action to take if oos_scores_table already exists. One of "fail", "append", "drop", or "truncate". The default is "fail".

model_name
Optional, the prefix of the Platform modeling jobs. It will have "Train" or "Predict" added to become the Script title.

cpu_requested
Optional, the number of CPU shares requested in the Civis Platform for training jobs or prediction child jobs. 1024 shares = 1 CPU.

memory_requested
Optional, the memory requested from Civis Platform for training jobs or prediction child jobs, in MiB.

disk_requested
Optional, the disk space requested on Civis Platform for training jobs or prediction child jobs, in GB.

notifications
Optional, model status notifications. See scripts_post_custom for further documentation about email and URL notification.

polling_interval
Check for job completion every this number of seconds.

verbose
Optional, If TRUE, supply debug outputs in Platform logs and make prediction child jobs visible.

civisml_version
Optional, a one-length character vector of the CivisML version. The default is "prod", the latest version in production

Value
A civis_ml object, a list containing the following elements:

job
job metadata from scripts_get_custom.

run
run metadata from scripts_get_custom_runs.

outputs
CivisML metadata from scripts_list_custom_runs_outputs containing the locations of files produced by CivisML e.g. files, projects, metrics, model_info, logs, predictions, and estimators.

metrics
Parsed CivisML output from metrics.json containing metadata from validation. A list containing the following elements:

• run list, metadata about the run.
• data list, metadata about the training data.
• model list, the fitted scikit-learn model with CV results.
• metrics list, validation metrics (accuracy, confusion, ROC, AUC, etc).
• warnings list.
• data_platform list, training data location.
civis_ml_sparse_ridge_regressor

model_info

Parsed CivisML output from model_info.json containing metadata from training. A list containing the following elements:

• run list, metadata about the run.
• data list, metadata about the training data.
• model list, the fitted scikit-learn model.
• metrics empty list.
• warnings list.
• data_platform list, training data location.

Data Sources

For building models with civis_ml, the training data can reside in four different places, a file in the Civis Platform, a CSV or feather-format file on the local disk, a data.frame resident in local the R environment, and finally, a table in the Civis Platform. Use the following helpers to specify the data source when calling civis_ml:

data.frame civis_ml(x = df,...)
local csv file civis_ml(x = "path/to/data.csv",...)
file in Civis Platform civis_ml(x = civis_file(1234))
table in Civis Platform civis_ml(x = civis_table(table_name = "schema.table",database_name = "database"))

Examples

## Not run:
data(ChickWeight)
m <- civis_ml_sparse_ridge_regressor(ChickWeight, dependent_variable = "weight", alpha = 999)
yhat <- fetch_oos_scores(m)

# Grid search
cv_params <- list(alpha = c(.001, .01, .1, 1))
m <- civis_ml_sparse_ridge_regressor(ChickWeight, dependent_variable = "weight",
cross_validation_parameters = cv_params,
calibration = NULL)

# make a prediction job, storing in a redshift table
pred_info <- predict(m, newdata = civis_table("schema.table", "my_database"),
output_table = "schema.scores_table")

## End(Not run)
civis_platform

Evaluate an expression in Civis Platform

Description
This is used as with the future API as an argument to plan.

Usage
civis_platform(...)

Arguments
... Arguments to CivisFuture and then scripts_post_containers

Value
The result of evaluating expr.

Examples
```r
## Not run:
plan(civis_platform)

# Specify required resources, image, and tag.
fut <- future({2 + 2},
  required_resources = list(cpu = 1024, memory = 2048),
  docker_image_name = "civisanalytics/datascience-r",
  docker_image_tag = "3")

# Evaluate the future later
fut <- future({2 + 2}, lazy = TRUE)
run(fut)

# check if a future has resolved
resolved(fut)

# block until the future resolves, and return the value or throw error
value(fut)

# cancel the job
cancel(fut)

# grab the run logs
fetch_logs(fut)

# handle errors
fut <- future({stop("Error!")})
e <- tryCatch(value(fut), error = function(e) e)
```
get_error(e)

## End(Not run)

civis_script  

Description

Create a civis_script object from a job and run id.

Usage

civis_script(id, run_id = NULL)

Arguments

id   integer, job id of the script.
run_id   integer, id of the run. If NULL (default), the most recent run will be used.

Details

A civis_script can be any custom, container, R, Python, SQL, or Javascript platform script.

Value

a civis_script object.

See Also

read_civis

Other script utils: fetch_output_file_ids(), run_civis(), run_template()
civis_table

A table in the Civis Platform

Description

Use `civis_table` to use a table in the Civis Platform with `civis_ml`.

Usage

```r
  civis_table(
    table_name,
    database_name = NULL,
    sql_where = NULL,
    sql_limit = NULL
  )
```

Arguments

- `table_name` The table name, as "schema.table".
- `database_name` The name of the database holding `table_name`.
- `sql_where` A SQL WHERE clause used to scope the rows of the training or prediction table. Note, the WHERE keyword is added at runtime.
- `sql_limit` A SQL LIMIT clause for querying the training or prediction set. Note, the LIMIT keyword is added at runtime.

Value

A `civis_table`.

civis_to_multifile_csv

Split a Redshift table into multiple CSV parts on S3

Description

Split a Redshift table into multiple CSV parts on S3 and return their locations as file IDs and presigned S3 urls.
civis_to_multifile_csv

Usage
civis_to_multifile_csv(
    sql,
    database,
    job_name = NULL,
    hidden = TRUE,
    include_header = TRUE,
    compression = "gzip",
    delimiter = ",",
    unquoted = FALSE,
    prefix = NULL
)

Arguments

sql string, The SQL select string to be executed.
database string, Name of database where query is run.
job_name string, optional. Name to identify scripted sql job.
hidden logical, Whether to hide the query in platform.
include_header logical, optional. Whether to include headers as an element in the returned list.
compression string, optional, Type of compression to use, if any. One of 'none', 'zip', or 'gzip'.
delimiter string, optional. Which delimiter to use. One of ',', '\t' or '|'.
unquoted logical, optional. Whether or not to quote fields.
prefix string, optional. A user specified filename prefix for the output files to have.

Details

When tables are large, unloading by splitting them first is faster. When we split a table, each Redshift compute node can dump its data into S3 in parallel with the others. By doing so, we avoid having all compute nodes sending the data through the leader node, which is slow.

This function returns a list that contains the location of the CSV parts as file IDs and presigned S3 urls. The user can use either the file IDs or the presigned S3 urls to download the CSV parts. The content of the list returned by this function is similar to that of the manifest file returned by Amazon S3 UNLOAD statements.

Value

A list with the items:

- header: column headers if 'include_header' is TRUE
- query: the executed query
- entries: a list containing presigned urls for each csv part
- compression: the type of compression on each csv part
- delimiter: the delimiter used to separate fields
- unquoted: whether fields are quoted
Examples

```r
## Not run:
# Download a table into multiple csv parts
sql <- "SELECT * FROM schema.table"
database <- "important_database"
manifest <- civis_to_multifile_csv(sql=sql, database=database)
files <- lapply(manifest["entries"], function(x) {
  download_civis(x$id, x$name)
  x$name
})

## End(Not run)
```

---

`clusters_delete_kubernetes_partitions`

*Delete a Cluster Partition*

**Description**

Delete a Cluster Partition

**Usage**

`clusters_delete_kubernetes_partitions(id, cluster_partition_id)`

**Arguments**

- `id` integer required. The ID of the cluster which this partition belongs to.
- `cluster_partition_id` integer required. The ID of this cluster partition.

**Value**

An empty HTTP response

---

`clusters_get_kubernetes`

*Describe a Kubernetes Cluster*

**Description**

Describe a Kubernetes Cluster

**Usage**

`clusters_get_kubernetes(id, include_usage_stats = NULL)`
Arguments

id  integer required.
include_usage_stats
    boolean optional. When true, usage stats are returned in instance config objects. Defaults to false.

Value

A list containing the following elements:

id  integer, The ID of this cluster.
organizationId  string, The id of this cluster’s organization.
organizationName  string, The name of this cluster’s organization.
organizationSlug  string, The slug of this cluster’s organization.
clusterPartitions  array, An array containing the following fields:
    • clusterPartitionId  integer, The ID of this cluster partition.
    • name  string, The name of the cluster partition.
    • labels  array, Labels associated with this partition.
    • instanceConfigs  array, The instances configured for this cluster partition.
    • defaultInstanceConfigId  integer, The id of the InstanceConfig that is the default for this partition.
isNatEnabled  boolean, Whether this cluster needs a NAT gateway or not.
hours  number, The number of hours used this month for this cluster

clusters_get_kubernetes_instance_configs

Describe an Instance Config

Description

Describe an Instance Config

Usage

clusters_get_kubernetes_instance_configs(
    instance_config_id,
    include_usage_stats = NULL
)
Arguments

instance_config_id
integer required. The ID of this instance config.

include_usage_stats
boolean optional. When true, usage stats are returned in instance config objects. Defaults to false.

Value

A list containing the following elements:

instanceConfigId
integer, The ID of this InstanceConfig.

instanceType
string, An EC2 instance type. Possible values include t2.large, m4.xlarge, m4.2xlarge, m4.4xlarge, m5.12xlarge, and p2.xlarge.

minInstances
integer, The minimum number of instances of that type in this cluster.

maxInstances
integer, The maximum number of instances of that type in this cluster.

instanceMaxMemory
integer, The amount of memory (RAM) available to a single instance of that type in megabytes.

instanceMaxCpu
integer, The number of processor shares available to a single instance of that type in millicores.

instanceMaxDisk
integer, The amount of disk available to a single instance of that type in gigabytes.

usageStats
list, A list containing the following elements:

• pendingMemoryRequested integer, The sum of memory requests (in MB) for pending deployments in this instance config.
• pendingCpuRequested integer, The sum of cpu requests (in millicores) for pending deployments in this instance config.
• runningMemoryRequested integer, The sum of memory requests (in MB) for running deployments in this instance config.
• runningCpuRequested integer, The sum of cpu requests (in millicores) for running deployments in this instance config.
• pendingDeployments integer, The number of pending deployments in this instance config.
• runningDeployments integer, The number of running deployments in this instance config.
clusters_get_kubernetes_partitions

Describe a Cluster Partition

Usage

clusters_get_kubernetes_partitions(
    id,
    cluster_partition_id,
    include_usage_stats = NULL
)

Arguments

id: integer required. The ID of the cluster which this partition belongs to.
cluster_partition_id: integer required. The ID of this cluster partition.
include_usage_stats: boolean optional. When true, usage stats are returned in instance config objects. Defaults to false.

Value

A list containing the following elements:

- clusterPartitionId: integer, The ID of this cluster partition.
- name: string, The name of the cluster partition.
- labels: array, Labels associated with this partition.
- instanceConfigs: array, An array containing the following fields:
  - instanceConfigId: integer, The ID of this InstanceConfig.
  - instanceType: string, An EC2 instance type. Possible values include t2.large, m4.xlarge, m4.2xlarge, m4.4xlarge, m5.12xlarge, and p2.xlarge.
  - minInstances: integer, The minimum number of instances of that type in this cluster.
  - maxInstances: integer, The maximum number of instances of that type in this cluster.
  - instanceMaxMemory: integer, The amount of memory (RAM) available to a single instance of that type in megabytes.
  - instanceMaxCpu: integer, The number of processor shares available to a single instance of that type in millicores.
clusters_list_kubernetes

**List Kubernetes Clusters**

**Description**  
List Kubernetes Clusters

**Usage**

```python
clusters_list_kubernetes(
    organization_slug = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```

**Arguments**

- `organization_slug`  
  string optional. The slug of this cluster’s organization.

- `limit`  
  integer optional. Number of results to return. Defaults to its maximum of 50.

- `page_num`  
  integer optional. Page number of the results to return. Defaults to the first page, 1.

- `order`  
  string optional. The field on which to order the result set. Defaults to organization_id. Must be one of: organization_id, created_at.

- `order_dir`  
  string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

**Value**

An array containing the following fields:

- `id`  
  integer. The ID of this cluster.

- `organizationId`  
  string. The id of this cluster’s organization.

- `organizationName`  
  string. The name of this cluster’s organization.

- `organizationSlug`  
  string. The slug of this cluster’s organization.
clusters_list_kubernetes_deployments

ClusterPartitions array, An array containing the following fields:
- clusterPartitionId integer, The ID of this cluster partition.
- name string, The name of the cluster partition.
- labels array, Labels associated with this partition.
- instanceConfigs array, The instances configured for this cluster partition.
- defaultInstanceConfigId integer, The id of the InstanceConfig that is the default for this partition.

isNatEnabled boolean, Whether this cluster needs a NAT gateway or not.

clusters_list_kubernetes_deployments

List the deployments associated with a Kubernetes Cluster

Description

List the deployments associated with a Kubernetes Cluster

Usage

clusters_list_kubernetes_deployments(
  id,
  base_type = NULL,
  state = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)

Arguments

id integer required. The id of the cluster.
base_type string optional. If specified, return deployments of these base types. It accepts a comma-separated list, possible values are 'Notebook', 'Service', 'Run'.
state string optional. If specified, return deployments in these states. It accepts a comma-separated list, possible values are pending, running, terminated, sleeping
limit integer optional. Number of results to return. Defaults to its maximum of 50.
page_num integer optional. Page number of the results to return. Defaults to the first page, 1.
order string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at.
order_dir string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.
clusters_list_kubernetes_deployment_stats

Value

An array containing the following fields:

- id: integer, The id of this deployment.
- name: string, The name of the deployment.
- baseId: integer, The id of the base object associated with the deployment.
- baseType: string, The base type of this deployment.
- state: string, The state of the deployment.
- cpu: integer, The CPU in millicores required by the deployment.
- memory: integer, The memory in MB required by the deployment.
- diskSpace: integer, The disk space in GB required by the deployment.
- instanceType: string, The EC2 instance type requested for the deployment.
- author: list, A list containing the following elements:
  - id: integer, The ID of this user.
  - name: string, This user’s name.
  - username: string, This user’s username.
  - initials: string, This user’s initials.
  - online: boolean, Whether this user is online.
- createdAt: string,
- updatedAt: string,

clusters_list_kubernetes_deployment_stats

Get stats about deployments associated with a Kubernetes Cluster

Description

Get stats about deployments associated with a Kubernetes Cluster

Usage

clusters_list_kubernetes_deployment_stats(id)

Arguments

id: integer required. The ID of this cluster.
clusters_list_kubernetes_instance_configs_historical_graphs

Value

An array containing the following fields:

- `baseType`: string, The base type of this deployment
- `state`: string, State of the deployment
- `count`: integer, Number of deployments of base type and state
- `totalCpu`: integer, Total amount of CPU in millicores for deployments of base type and state
- `totalMemory`: integer, Total amount of Memory in megabytes for deployments of base type and state

clusters_list_kubernetes_instance_configs_historical_graphs

Get graphs of historical resource usage in an Instance Config

Description

Get graphs of historical resource usage in an Instance Config

Usage

```
clusters_list_kubernetes_instance_configs_historical_graphs(
    instance_config_id,
    timeframe = NULL
)
```

Arguments

- `instance_config_id`: integer required. The ID of this instance config.
- `timeframe`: string optional. The span of time that the graphs cover. Must be one of 1_day, 1_week.

Value

A list containing the following elements:

- `cpuGraphUrl`: string, URL for the graph of historical CPU usage in this instance config.
- `memGraphUrl`: string, URL for the graph of historical memory usage in this instance config.
clusters_list_kubernetes_instance_configs_user_statistics

Get statistics about the current users of an Instance Config

Description

Get statistics about the current users of an Instance Config

Usage

clusters_list_kubernetes_instance_configs_user_statistics(
  instance_config_id,
  order = NULL,
  order_dir = NULL
)

Arguments

instance_config_id
  integer required. The ID of this instance config.

order
  string optional. The field on which to order the result set. Defaults to running_deployments. Must be one of pending_memory_requested, pending_cpu_requested, running_memory_requested, running_cpu_requested, pending_deployments, running_deployments.

order_dir
  string optional. Direction in which to sort, either asc (ascending) or desc (descending). Defaults to desc.

Value

An array containing the following fields:

userId
  string, The owning user's ID

userName
  string, The owning user's name

pendingDeployments
  integer, The number of deployments belonging to the owning user in "pending" state

pendingMemoryRequested
  integer, The sum of memory requests (in MB) for deployments belonging to the owning user in "pending" state

pendingCpuRequested
  integer, The sum of CPU requests (in millicores) for deployments belonging to the owning user in "pending" state

runningDeployments
  integer, The number of deployments belonging to the owning user in "running" state
clusters_list_kubernetes_partitions

runningMemoryRequested
integer, The sum of memory requests (in MB) for deployments belonging to the owning user in "running" state

runningCpuRequested
integer, The sum of CPU requests (in millicores) for deployments belonging to the owning user in "running" state

clusters_list_kubernetes_partitions

List Cluster Partitions for given cluster

Description
List Cluster Partitions for given cluster

Usage
clusters_list_kubernetes_partitions(id, include_usage_stats = NULL)

Arguments
id
integer required.
include_usage_stats
boolean optional. When true, usage stats are returned in instance config objects. Defaults to false.

Value
An array containing the following fields:

clusterPartitionId
integer, The ID of this cluster partition.

name
string, The name of the cluster partition.

labels
array, Labels associated with this partition.

instanceConfigs
array, An array containing the following fields:

• instanceConfigId integer, The ID of this InstanceConfig.
• instanceType string, An EC2 instance type. Possible values include t2.large, m4.xlarge, m4.2xlarge, m4.4xlarge, m5.12xlarge, and p2.xlarge.
• minInstances integer, The minimum number of instances of that type in this cluster.
• maxInstances integer, The maximum number of instances of that type in this cluster.
• instanceMaxMemory integer, The amount of memory (RAM) available to a single instance of that type in megabytes.
clusters_patch_kubernetes

**Update a Kubernetes Cluster**

**Description**

Update a Kubernetes Cluster

**Usage**

```python
clusters_patch_kubernetes(id, is_nat_enabled = NULL)
```

**Arguments**

- `id` integer required. The ID of this cluster.
- `is_nat_enabled` boolean optional. Whether this cluster needs a NAT gateway or not.

**Value**

A list containing the following elements:

- `id` integer, The ID of this cluster.
- `organizationId` string, The id of this cluster’s organization.
- `organizationName` string, The name of this cluster’s organization.
- `organizationSlug` string, The slug of this cluster’s organization.
- `clusterPartitions` array, An array containing the following fields:
  - `clusterPartitionId` integer, The ID of this cluster partition.
  - `name` string, The name of the cluster partition.
  - `labels` array, Labels associated with this partition.
  - `instanceConfigs` array, The instances configured for this cluster partition.
  - `defaultInstanceConfigId` integer, The id of the InstanceConfig that is the default for this partition.
- `isNatEnabled` boolean, Whether this cluster needs a NAT gateway or not.
- `hours` number, The number of hours used this month for this cluster.
clusters_patch_kubernetes_partitions

Update a Cluster Partition

Description

Update a Cluster Partition

Usage

clusters_patch_kubernetes_partitions(
    id,
    cluster_partition_id,
    instance_configs = NULL,
    name = NULL,
    labels = NULL
)

Arguments

id    integer required. The ID of the cluster which this partition belongs to.

cluster_partition_id    integer required. The ID of this cluster partition.

instance_configs    array optional. An array containing the following fields:
    • instanceType string, An EC2 instance type. Possible values include t2.large, m4.xlarge, m4.2xlarge, m4.4xlarge, m5.12xlarge, and p2.xlarge.
    • minInstances integer, The minimum number of instances of that type in this cluster.
    • maxInstances integer, The maximum number of instances of that type in this cluster.

name    string optional. The name of the cluster partition.

labels    array optional. Labels associated with this partition.

Value

A list containing the following elements:

clusterPartitionId    integer, The ID of this cluster partition.

name    string, The name of the cluster partition.

labels    array, Labels associated with this partition.

instanceConfigs    array, An array containing the following fields:
    • instanceConfigId integer, The ID of this InstanceConfig.
clusters_post_kubernetes

Create a Kubernetes Cluster

Description

Create a Kubernetes Cluster

Usage

clusters_post_kubernetes(
    organization_id = NULL,
    organization_slug = NULL,
    is_nat_enabled = NULL
)

Arguments

organization_id
    string optional. The id of this cluster’s organization.

organization_slug
    string optional. The slug of this cluster’s organization.

is_nat_enabled    boolean optional. Whether this cluster needs a NAT gateway or not.

Value

A list containing the following elements:

id          integer, The ID of this cluster.

organizationId  string, The id of this cluster’s organization.
organizationName
   string, The name of this cluster's organization.

organizationSlug
   string, The slug of this cluster's organization.

clusterPartitions
   array, An array containing the following fields:
   - clusterPartitionId integer, The ID of this cluster partition.
   - name string, The name of the cluster partition.
   - labels array, Labels associated with this partition.
   - instanceConfigs array, The instances configured for this cluster partition.
   - defaultInstanceConfigId integer, The id of the InstanceConfig that is the default for this partition.

isNatEnabled
   boolean, Whether this cluster needs a NAT gateway or not.

hours
   number, The number of hours used this month for this cluster

---

clusters_post_kubernetes_partitions

Create a Cluster Partition for given cluster

Description

Create a Cluster Partition for given cluster

Usage

clusters_post_kubernetes_partitions(id, instance_configs, name, labels)

Arguments

id
   integer required. The ID of the cluster which this partition belongs to.

instance_configs
   array required. An array containing the following fields:
   - instanceType string, An EC2 instance type. Possible values include t2.large, m4.xlarge, m4.2xlarge, m4.4xlarge, m5.12xlarge, and p2.xlarge.
   - minInstances integer, The minimum number of instances of that type in this cluster.
   - maxInstances integer, The maximum number of instances of that type in this cluster.

name
   string required. The name of the cluster partition.

labels
   array required. Labels associated with this partition.
Value

A list containing the following elements:

- `clusterPartitionId` integer, The ID of this cluster partition.
- `name` string, The name of the cluster partition.
- `labels` array, Labels associated with this partition.
- `instanceConfigs` array, An array containing the following fields:
  - `instanceConfigId` integer, The ID of this InstanceConfig.
  - `instanceType` string, An EC2 instance type. Possible values include t2.large, m4.xlarge, m4.2xlarge, m4.4xlarge, m5.12xlarge, and p2.xlarge.
  - `minInstances` integer, The minimum number of instances of that type in this cluster.
  - `maxInstances` integer, The maximum number of instances of that type in this cluster.
  - `instanceMaxMemory` integer, The amount of memory (RAM) available to a single instance of that type in megabytes.
  - `instanceMaxCpu` integer, The number of processor shares available to a single instance of that type in millicores.
  - `instanceMaxDisk` integer, The amount of disk available to a single instance of that type in gigabytes.
  - `usageStats` object,
- `defaultInstanceConfigId` integer, The id of the InstanceConfig that is the default for this partition.

---

`coef.civis_ml`  
*Get civis_ml model coefficients*

Description

Get civis_ml model coefficients

Usage

```r
## S3 method for class 'civis_ml'
coef(object, complete = TRUE, ...)
```

Arguments

- `object` civis_ml_model
- `complete` see documentation for generic `coef()`
- `...` other arguments
credentials_delete_shares_groups

Details
Outputs coefficients with names in the style of ‘stats::coef’

Value
a matrix of coefficients, or ‘NULL’ if none available from CivisML

Description
Revoke the permissions a group has on this object

Usage
credentials_delete_shares_groups(id, group_id)

Arguments
id integer required. The ID of the resource that is shared.
group_id integer required. The ID of the group.

Value
An empty HTTP response

credentials_delete_shares_users

Description
Revoke the permissions a user has on this object

Usage
credentials_delete_shares_users(id, user_id)

Arguments
id integer required. The ID of the resource that is shared.
user_id integer required. The ID of the user.

Value
An empty HTTP response
credentials_get

Get a credential

Description
Get a credential

Usage
credentials_get(id)

Arguments
id integer required. The ID of the credential.

Value
A list containing the following elements:

id integer, The ID of the credential.
name string, The name identifying the credential
type string, The credential’s type.
username string, The username for the credential.
description string, A long description of the credential.
owner string, The name of the user who this credential belongs to.
remoteHostId integer, The ID of the remote host associated with this credential.
remoteHostName string, The name of the remote host associated with this credential.
state string, The U.S. state for the credential. Only for VAN credentials.
createdAt string, The creation time for this credential.
updatedAt string, The last modification time for this credential.

credentials_list

List credentials

Description
List credentials
Usage

credentials_list(
    type = NULL,
    remote_host_id = NULL,
    default = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)

Arguments

type string optional. The type (or types) of credentials to return. One or more of: Amazon Web Services S3, Bitbucket, CASS/NCOA PAF, Certificate, Civis Platform, Custom, Database, Google, Github, Salesforce User, Salesforce Client, and TableauUser. Specify multiple values as a comma-separated list (e.g., "A,B").

remote_host_id integer optional. The ID of the remote host associated with the credentials to return.

default boolean optional. If true, will return a list with a single credential which is the current user’s default credential.

limit integer optional. Number of results to return. Defaults to its maximum of 1000.

page_num integer optional. Page number of the results to return. Defaults to the first page, 1.

order string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, created_at, name.

order_dir string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id integer, The ID of the credential.

name string, The name identifying the credential

type string, The credential’s type.

username string, The username for the credential.

description string, A long description of the credential.

owner string, The name of the user who this credential belongs to.

remoteHostId integer, The ID of the remote host associated with this credential.

remoteHostName string, The name of the remote host associated with this credential.

state string, The U.S. state for the credential. Only for VAN credentials.

createdAt string, The creation time for this credential.

updatedAt string, The last modification time for this credential.
credentials_list_shares

List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

credentials_list_shares(id)

Arguments

id integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers list, A list containing the following elements:
  • users array,
  • groups array,

writers list, A list containing the following elements:
  • users array,
  • groups array,

owners list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.
credentials_post

Create a credential

Description
Create a credential

Usage
credentials_post(
  type,
  username,
  password,
  name = NULL,
  description = NULL,
  remote_host_id = NULL,
  state = NULL,
  system_credential = NULL
)

Arguments
- type: string required.
- username: string required. The username for the credential.
- password: string required. The password for the credential.
- name: string optional. The name identifying the credential.
- remote_host_id: integer optional. The ID of the remote host associated with the credential.
- system_credential: boolean optional.

Value
A list containing the following elements:
- id: integer, The ID of the credential.
- name: string, The name identifying the credential.
- type: string, The credential’s type.
- username: string, The username for the credential.
- description: string, A long description of the credential.
- owner: string, The name of the user who this credential belongs to.
- remoteHostId: integer, The ID of the remote host associated with this credential.
remoteHostId string, The ID of the remote host associated with this credential.
remoteHostName string, The name of the remote host associated with this credential.
state string, The U.S. state for the credential. Only for VAN credentials.
createdAt string, The creation time for this credential.
updatedAt string, The last modification time for this credential.

credentials_post_authenticate

Authenticate against a remote host

Description
Authenticate against a remote host

Usage
credentials_post_authenticate(url, remote_host_type, username, password)

Arguments
url string required. The URL to your host.
remote_host_type string required. The type of remote host. One of: RemoteHostTypes::Bitbucket, RemoteHostTypes::GitSSH, RemoteHostTypes::Github, RemoteHostTypes::GoogleDoc, RemoteHostTypes::JDBC, RemoteHostTypes::Postgres, RemoteHostTypes::Redshift, RemoteHostTypes::S3Storage, and RemoteHostTypes::Salesforce
username string required. The username for the credential.
password string required. The password for the credential.

Value
A list containing the following elements:
id integer, The ID of the credential.
name string, The name identifying the credential
type string, The credential’s type.
username string, The username for the credential.
description string, A long description of the credential.
owner string, The name of the user who this credential belongs to.
remoteHostId integer, The ID of the remote host associated with this credential.
remoteHostName string, The name of the remote host associated with this credential.
state string, The U.S. state for the credential. Only for VAN credentials.
createdAt string, The creation time for this credential.
updatedAt string, The last modification time for this credential.
credentials_post_temporary

Generate a temporary credential for accessing S3

Description
Generate a temporary credential for accessing S3

Usage
credentials_post_temporary(id, duration = NULL)

Arguments
- id: integer required. The ID of the credential.
- duration: integer optional. The number of seconds the temporary credential should be valid. Defaults to 15 minutes. Must not be less than 15 minutes or greater than 36 hours.

Value
A list containing the following elements:
- accessKey: string, The identifier of the credential.
- secretAccessKey: string, The secret part of the credential.
- sessionToken: string, The session token identifier.

credentials_put
Update an existing credential

Description
Update an existing credential

Usage
credentials_put(
  id,
  type,
  username,
  password,
  name = NULL,
  description = NULL,
  remote_host_id = NULL,
  state = NULL,
  system_credential = NULL
)

Arguments

* id integer required. The ID of the credential.
* type string required.
* username string required. The username for the credential.
* password string required. The password for the credential.
* name string optional. The name identifying the credential.
* description string optional. A long description of the credential.
* remote_host_id integer optional. The ID of the remote host associated with the credential.
* state string optional. The U.S. state for the credential. Only for VAN credentials.
* system_credential boolean optional.

Value

A list containing the following elements:

* id integer, The ID of the credential.
* name string, The name identifying the credential
* type string, The credential’s type.
* username string, The username for the credential.
* description string, A long description of the credential.
* owner string, The name of the user who this credential belongs to.
* remoteHostId integer, The ID of the remote host associated with this credential.
* remoteHostName string, The name of the remote host associated with this credential.
* state string, The U.S. state for the credential. Only for VAN credentials.
* createdAt string, The creation time for this credential.
* updatedAt string, The last modification time for this credential.

---

credentials_put_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

credentials_put_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
credentials_put_shares_users

Arguments

- **id**
  integer required. The ID of the resource that is shared.

- **group_ids**
  array required. An array of one or more group IDs.

- **permission_level**
  string required. Options are: "read", "write", or "manage".

- **share_email_body**
  string optional. Custom body text for e-mail sent on a share.

- **send_shared_email**
  boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- **readers**
  list, A list containing the following elements:
  - users array,
  - groups array,

- **writers**
  list, A list containing the following elements:
  - users array,
  - groups array,

- **owners**
  list, A list containing the following elements:
  - users array,
  - groups array,

- **totalUserShares**
  integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

- **totalGroupShares**
  integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

credentials_put_shares_users

*Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

credentials_put_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
databases_delete_whitelist_ips

Remove a whitelisted IP address

Arguments

id integer required. The ID of the resource that is shared.
user_ids array required. An array of one or more user IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers list, A list containing the following elements:
  • users array,
  • groups array,

writers list, A list containing the following elements:
  • users array,
  • groups array,

owners list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

Description

Remove a whitelisted IP address

Usage

databases_delete_whitelist_ips(id, whitelisted_ip_id)
Arguments

   id    integer required. The ID of the database this rule is applied to.
whitelisted_ip_id    integer required. The ID of this whitelisted IP address.

Value

An empty HTTP response

databases_get

Show database information

Description

Show database information

Usage

databases_get(id)

Arguments

id    integer required. The ID for the database.

Value

A list containing the following elements:

   id    integer, The ID for the database.
   name    string, The name of the database.
   adapter    string, The type of the database.

databases_get_whitelist_ips

View details about a whitelisted IP

Description

View details about a whitelisted IP

Usage

databases_get_whitelist_ips(id, whitelisted_ip_id)
Arguments

- **id**: integer required. The ID of the database this rule is applied to.
- **whitelisted_ip_id**: integer required. The ID of this whitelisted IP address.

Value

A list containing the following elements:

- **id**: integer, The ID of this whitelisted IP address.
- **remoteHostId**: integer, The ID of the database this rule is applied to.
- **securityGroupId**: string, The ID of the security group this rule is applied to.
- **subnetMask**: string, The subnet mask that is allowed by this rule.
- **authorizedBy**: string, The user who authorized this rule.
- **isActive**: boolean, True if the rule is applied, false if it has been revoked.
- **createdAt**: string, The time this rule was created.
- **updatedAt**: string, The time this rule was last updated.

---

`databases_list`  
*List databases*

### Description

List databases

### Usage

`databases_list()`

### Value

An array containing the following fields:

- **id**: integer, The ID for the database.
- **name**: string, The name of the database.
- **adapter**: string, The type of the database.
**databases_list_advanced_settings**

*Get the advanced settings for this database*

**Description**

Get the advanced settings for this database

**Usage**

databases_list_advanced_settings(id)

**Arguments**

id  
integer required. The ID of the database this advanced settings object belongs to.

**Value**

A list containing the following elements:

- exportCachingEnabled
  - boolean, Whether or not caching is enabled for export jobs run on this database server.

---

**databases_list_schemas**

*List schemas in this database*

**Description**

List schemas in this database

**Usage**

databases_list_schemas(id)

**Arguments**

id  
integer required. The ID of the database.

**Value**

An array containing the following fields:

- schema
  - string, The name of a schema.
databases_list_tables  List tables in the specified database, deprecated use "GET /tables" instead

Description
List tables in the specified database, deprecated use "GET /tables" instead

Usage
databases_list_tables(
    id,
    name = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)

Arguments
id  integer required. The ID of the database.
name  string optional. If specified, will be used to filter the tables returned. Substring matching is supported (e.g., "name=table" will return both "table1" and "my table").
limit  integer optional. Number of results to return. Defaults to 200. Maximum allowed is 1000.
page_num  integer optional. Page number of the results to return. Defaults to the first page, 1.
order  string optional. The field on which to order the result set. Defaults to name. Must be one of: name.
order_dir  string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value
An array containing the following fields:
id  integer, The ID of the table.
databaseId  integer, The ID of the database.
schema  string, The name of the schema containing the table.
name  string, Name of the table.
description  string, The description of the table, as specified by the table owner
isView  boolean, True if this table represents a view. False if it represents a regular table.
databases_list_tables_search

List tables in the specified database, deprecated use "GET /tables" instead

Description

List tables in the specified database, deprecated use "GET /tables" instead

Usage

databases_list_tables_search(id, name = NULL, column_name = NULL)

Arguments

id integer required. The ID of the database.

name string optional. If specified, will be used to filter the tables returned. Substring matching is supported (e.g., "name=table" will return both "table1" and "my table").

column_name string optional. Search for tables containing a column with the given name.
Value

An array containing the following fields:

- **id**: integer, The ID of the table.
- **databaseId**: integer, The ID of the database.
- **schema**: string, The name of the schema containing the table.
- **name**: string, Name of the table.
- **description**: string, The description of the table, as specified by the table owner
- **isView**: boolean, True if this table represents a view. False if it represents a regular table.
- **rowCount**: integer, The number of rows in the table.
- **columnCount**: integer, The number of columns in the table.
- **sizeMb**: number, The size of the table in megabytes.
- **owner**: string, The database username of the table’s owner.
- **distkey**: string, The column used as the Amazon Redshift distkey.
- **sortkeys**: string, The column used as the Amazon Redshift sortkey.
- **refreshStatus**: string, How up-to-date the table’s statistics on row counts, null counts, distinct counts, and values distributions are. One of: refreshing, stale, or current.
- **lastRefresh**: string, The time of the last statistics refresh.
- **refreshId**: string, The ID of the most recent statistics refresh.
- **lastRun**: list, A list containing the following elements:
  - **id**: integer,
  - **state**: string,
  - **createdAt**: string, The time that the run was queued.
  - **startedAt**: string, The time that the run started.
  - **finishedAt**: string, The time that the run completed.
  - **error**: string, The error message for this run, if present.
- **columnName**: array, The names of each column in the table.

---

**databases_list_whitelist_ips**

List whitelisted IPs for the specified database

**Description**

List whitelisted IPs for the specified database

**Usage**

databases_list_whitelist_ips(id)
**Arguments**

id  integer required. The ID for the database.

**Value**

An array containing the following fields:

- **id**  integer, The ID of this whitelisted IP address.
- **remoteHostId**  integer, The ID of the database this rule is applied to.
- **securityGroupId**  string, The ID of the security group this rule is applied to.
- **subnetMask**  string, The subnet mask that is allowed by this rule.
- **createdAt**  string, The time this rule was created.
- **updatedAt**  string, The time this rule was last updated.

---

**databases_patch_advanced_settings**

*Update the advanced settings for this database*

**Description**

Update the advanced settings for this database

**Usage**

databases_patch_advanced_settings(id, export_caching_enabled = NULL)

**Arguments**

id  integer required. The ID of the database this advanced settings object belongs to.

export_caching_enabled  boolean optional. Whether or not caching is enabled for export jobs run on this database server.

**Value**

A list containing the following elements:

- **exportCachingEnabled**  boolean, Whether or not caching is enabled for export jobs run on this database server.
databases_post_schemas_scan

*Creates and enqueues a schema scanner job*

**Description**

Creates and enqueues a schema scanner job

**Usage**

```bash
databases_post_schemas_scan(id, schema, stats_priority = NULL)
```

**Arguments**

- **id** integer required. The ID of the database.
- **schema** string required. The name of the schema.
- **stats_priority** string optional. When to sync table statistics for every table in the schema. Valid options are the following. Option: 'flag' means to flag stats for the next scheduled run of a full table scan on the database. Option: 'block' means to block this job on stats syncing. Option: 'queue' means to queue a separate job for syncing stats and do not block this job on the queued job. Defaults to 'flag'.

**Value**

A list containing the following elements:

- **jobId** integer, The ID of the job created.
- **runId** integer, The ID of the run created.

---

databases_post_whitelist_ips

*Whitelist an IP address*

**Description**

Whitelist an IP address

**Usage**

```bash
databases_post_whitelist_ips(id, subnet_mask)
```

**Arguments**

- **id** integer required. The ID of the database this rule is applied to.
- **subnet_mask** string required. The subnet mask that is allowed by this rule.
**Value**

A list containing the following elements:

- **id** integer, The ID of this whitelisted IP address.
- **remoteHostId** integer, The ID of the database this rule is applied to.
- **securityGroupId** string, The ID of the security group this rule is applied to.
- **subnetMask** string, The subnet mask that is allowed by this rule.
- **authorizedBy** string, The user who authorized this rule.
- **isActive** boolean, True if the rule is applied, false if it has been revoked.
- **createdAt** string, The time this rule was created.
- **updatedAt** string, The time this rule was last updated.

---

**databases_put_advanced_settings**

*Edit the advanced settings for this database*

---

**Description**

Edit the advanced settings for this database

**Usage**

`databases_put_advanced_settings(id, export_caching_enabled)`

**Arguments**

- **id** integer required. The ID of the database this advanced settings object belongs to.
- **export_caching_enabled** boolean required. Whether or not caching is enabled for export jobs run on this database server.

**Value**

A list containing the following elements:

- **exportCachingEnabled** boolean, Whether or not caching is enabled for export jobs run on this database server.
**default_credential**  
*The current user’s default credential.*

**Description**  
The current user’s default credential.

**Usage**  
default_credential()

**Value**  
credential_id, id of the default credential for the current user.

---

**download_civis**  
*Download a table or a file from the Civis Platform to local disk*

**Description**  
download_civis downloads a file based on the type of its first argument, which can be a string “schema.table”, a SQL query sql(...), or a numeric file ID.

A table or a query from Redshift will be downloaded onto disk as a CSV. A file from Platform files endpoint will be downloaded as is.

A default database can be set using `options(civis.default_db = "my_database")`. If there is only one database available, this database will automatically be used as the default.

**Usage**  
download_civis(x, ...)

## S3 method for class 'character'
download_civis(
  x,
  database = NULL,
  file,
  overwrite = FALSE,
  progress = FALSE,
  split = FALSE,
  job_name = NULL,
  hidden = TRUE,
  verbose = FALSE,
  ...
)
## S3 method for class 'sql'

```r
download_civis(
  x,
  database = NULL,
  file,
  overwrite = FALSE,
  progress = FALSE,
  split = FALSE,
  job_name = NULL,
  hidden = TRUE,
  verbose = FALSE,
  ...
)
```

## S3 method for class 'numeric'

```r
download_civis(x, file, overwrite = FALSE, progress = FALSE, ...)
```

### Arguments

- `x` "schema.table", sql("query"), or a file id.
- `...` Currently ignored.
- `database` string, The database. If `NULL`, tries to use the default database.
- `file` string, The file to write to.
- `overwrite` logical, Whether to overwrite the existing file.
- `progress` logical, Whether to display a progress bar.
- `split` logical, Whether to download a big table by splitting it into multiple CSV parts first. See `civis_to_multifile_csv` for details.
- `job_name` string, Name of the job (default: "Civis Download Via R Client").
- `hidden` logical, Whether the job is hidden on Platform.
- `verbose` logical, Whether to print detailed updates of job status.

### Value

The file where the downloaded files or tables are written to. It is returned invisibly.

### Methods (by class)

- `character`: Download a table from Redshift to disk as CSV.
- `sql`: Download the result of a SQL query from Redshift to disk as CSV.
- `numeric`: Download a file from Platform files endpoint to disk.

### See Also

Other `io`: `query_civis_file()`, `query_civis()`, `read_civis()`, `write_civis_file()`, `write_civis()`
Examples

```r
## Not run:
# Download all columns in a single table into a CSV
download_civis("schema.table", database = "my_database",
               file = "~/Downloads/my_table.csv")

# Download data from a SQL select statement into a CSV
query <- sql("SELECT * FROM table JOIN other_table USING id WHERE var1 < 23")
download_civis(query, database = "my_database",
               file = "~/Downloads/my_table.csv")

# Set a default database
options(civis.default_db = "my_database")

# Download any file from the files endpoint.
file_id <- write_civis_file(df)
download_civis(file_id, file = "df.rds", progress = TRUE)
df2 <- readRDS("df.rds")
identical(df, df2)
## End(Not run)
```

---

### endpoints_list

*List API endpoints*

**Description**

List API endpoints

**Usage**

`endpoints_list()`

**Value**

An empty HTTP response

---

### enhancements_delete_cass_ncoa_projects

*Remove a CASS/NCOA Enhancement from a project*

**Description**

Remove a CASS/NCOA Enhancement from a project
**enhancements_delete_cass_ncoa_runs**

Usage

```
enhancements_delete_cass_ncoa_projects(id, project_id)
```

Arguments

- `id`  
  integer required. The ID of the CASS/NCOA Enhancement.

- `project_id`  
  integer required. The ID of the project.

Value

An empty HTTP response

---

**enhancements_delete_cass_ncoa_runs**

*Cancel a run*

Description

Cancel a run

Usage

```
enhancements_delete_cass_ncoa_runs(id, run_id)
```

Arguments

- `id`  
  integer required. The ID of the cass_ncoa.

- `run_id`  
  integer required. The ID of the run.

Value

An empty HTTP response

---

**enhancements_delete_cass_ncoa_shares_groups**

*Revoke the permissions a group has on this object*

Description

Revoke the permissions a group has on this object

Usage

```
enhancements_delete_cass_ncoa_shares_groups(id, group_id)
```
Arguments

- `id` integer required. The ID of the resource that is shared.
- `group_id` integer required. The ID of the group.

Value

An empty HTTP response

---

**enhancements_delete_cass_ncoa_shares_users**

*Revoke the permissions a user has on this object*

Description

Revoke the permissions a user has on this object

Usage

`enhancements_delete_cass_ncoa_shares_users(id, user_id)`

Arguments

- `id` integer required. The ID of the resource that is shared.
- `user_id` integer required. The ID of the user.

Value

An empty HTTP response

---

**enhancements_delete_civis_data_match_projects**

*Remove a Civis Data Match Enhancement from a project*

Description

Remove a Civis Data Match Enhancement from a project

Usage

`enhancements_delete_civis_data_match_projects(id, project_id)`

Arguments

- `id` integer required. The ID of the Civis Data Match Enhancement.
- `project_id` integer required. The ID of the project.
Value

An empty HTTP response

---

**enhancements_delete_civis_data_match_runs**

*Cancel a run*

**Description**

Cancel a run

**Usage**

`enhancements_delete_civis_data_match_runs(id, run_id)`

**Arguments**

- **id** integer required. The ID of the civis_data_match.
- **run_id** integer required. The ID of the run.

**Value**

An empty HTTP response

---

**enhancements_delete_civis_data_match_shares_groups**

*Revoke the permissions a group has on this object*

**Description**

Revoke the permissions a group has on this object

**Usage**

`enhancements_delete_civis_data_match_shares_groups(id, group_id)`

**Arguments**

- **id** integer required. The ID of the resource that is shared.
- **group_id** integer required. The ID of the group.

**Value**

An empty HTTP response
enhancements_delete_civis_data_match_shares_users

Revoke the permissions a user has on this object

Description
Revoke the permissions a user has on this object

Usage
enhancements_delete_civis_data_match_shares_users(id, user_id)

Arguments
id integer required. The ID of the resource that is shared.
user_id integer required. The ID of the user.

Value
An empty HTTP response

enhancements_delete_geocode_projects

Remove a Geocode Enhancement from a project

Description
Remove a Geocode Enhancement from a project

Usage
enhancements_delete_geocode_projects(id, project_id)

Arguments
id integer required. The ID of the Geocode Enhancement.
project_id integer required. The ID of the project.

Value
An empty HTTP response
**enhancements_delete_geocode_runs**

*Cancel a run*

**Description**

Cancel a run

**Usage**

`enhancements_delete_geocode_runs(id, run_id)`

**Arguments**

- `id` integer required. The ID of the geocode.
- `run_id` integer required. The ID of the run.

**Value**

An empty HTTP response

---

**enhancements_delete_geocode_shares_groups**

*Revoke the permissions a group has on this object*

**Description**

Revoke the permissions a group has on this object

**Usage**

`enhancements_delete_geocode_shares_groups(id, group_id)`

**Arguments**

- `id` integer required. The ID of the resource that is shared.
- `group_id` integer required. The ID of the group.

**Value**

An empty HTTP response
**enhancements_delete_geocode_shares_users**

*Revoke the permissions a user has on this object*

**Description**
Revoke the permissions a user has on this object

**Usage**
```
enhancements_delete_geocode_shares_users(id, user_id)
```

**Arguments**
- **id** integer required. The ID of the resource that is shared.
- **user_id** integer required. The ID of the user.

**Value**
An empty HTTP response

**enhancements_get_cass_ncoa**

*Get a CASS/NCOA Enhancement*

**Description**
Get a CASS/NCOA Enhancement

**Usage**
```
enhancements_get_cass_ncoa(id)
```

**Arguments**
- **id** integer required.

**Value**
A list containing the following elements:
- **id** integer, The ID for the enhancement.
- **name** string, The name of the enhancement job.
- **type** string, The type of the enhancement (e.g. CASS-NCOA)
- **createdAt** string, The time this enhancement was created.
updatedAt string, The time the enhancement was last updated.

author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

state string, The status of the enhancement’s last run

schedule list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

parentId integer, Parent ID that triggers this enhancement.

notifications list, A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • successEmailFromName string, Name from which success emails are sent; defaults to “Civis.”
  • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  • successOn boolean, If success email notifications are on.
  • failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

source list, A list containing the following elements:
  • databaseTable list. A list containing the following elements:
- schema string, The schema name of the source table.
- table string, The name of the source table.
- remoteHostId integer, The ID of the database host for the table.
- credentialId integer, The id of the credentials to be used when performing the enhancement.
- multipartKey array, The source table primary key.

destination list, A list containing the following elements:
  • databaseTable list. A list containing the following elements:
    – schema string, The schema name for the output data.
    – table string, The table name for the output data.

columnMapping list, A list containing the following elements:
  • address1 string, The first address line.
  • address2 string, The second address line.
  • city string, The city of an address.
  • state string, The state of an address.
  • zip string, The zip code of an address.
  • name string, The full name of the resident at this address. If needed, separate multiple columns with ‘+’, e.g. 'first_name+last_name'
  • company string, The name of the company located at this address.

useDefaultColumnMapping boolean, Defaults to true, where the existing column mapping on the input table will be used. If false, a custom column mapping must be provided.

performNcoa boolean, Whether to update addresses for records matching the National Change of Address (NCOA) database.

ncoaCredentialId integer, Credential to use when performing NCOA updates. Required if 'performNcoa' is true.

outputLevel string, The set of fields persisted by a CASS or NCOA enhancement. For CASS enhancements, one of 'cass' or 'all'. For NCOA enhancements, one of 'cass', 'ncoa', 'coalesced' or 'all'. By default, all fields will be returned.

limitingSQL string, The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').

archived string, The archival status of the requested item(s).

---

**enhancements_get_cass_ncoa_runs**

*Check status of a run*

**Description**

Check status of a run
enhancements_get_civis_data_match

Usage

enhancements_get_cass_ncoa_runs(id, run_id)

Arguments

id integer required. The ID of the cass_ncoa.
run_id integer required. The ID of the run.

Value

A list containing the following elements:

id integer, The ID of the run.
cassNcoaId integer, The ID of the cass_ncoa.
state string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested boolean, True if run cancel requested, else false.
startedAt string, The time the last run started at.
finishedAt string, The time the last run completed.
error string, The error, if any, returned by the run.

enhancements_get_civis_data_match

Get a Civis Data Match Enhancement

Description

Get a Civis Data Match Enhancement

Usage

enhancements_get_civis_data_match(id)

Arguments

id integer required.
**Value**

A list containing the following elements:

- **id** integer, The ID for the enhancement.
- **name** string, The name of the enhancement job.
- **type** string, The type of the enhancement (e.g. CASS-NCOA)
- **createdAt** string, The time this enhancement was created.
- **updatedAt** string, The time the enhancement was last updated.
- **author** list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.
- **state** string, The status of the enhancement’s last run
- **schedule** list, A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.
- **parentId** integer, Parent ID that triggers this enhancement.
- **notifications** list, A list containing the following elements:
  - urls array, URLs to receive a POST request at job completion
  - successEmailSubject string, Custom subject line for success e-mail.
  - successEmailBody string, Custom body text for success e-mail, written in Markdown.
  - successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  - successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  - successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  - failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  - stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  - successOn boolean, If success email notifications are on.
  - failureOn boolean, If failure email notifications are on.
- **runningAs** list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

inputFieldMapping
list, The column mapping for the input table. See /enhancements/field_mapping for list of valid fields.

inputTable
list, A list containing the following elements:
• databaseName string, The Redshift database name for the table.
• schema string, The schema name for the table.
• table string, The table name.

matchTargetId
integer, The ID of the Civis Data match target. See /match_targets for IDs.

outputTable
list, A list containing the following elements:
• databaseName string, The Redshift database name for the table.
• schema string, The schema name for the table.
• table string, The table name.

maxMatches
integer, The maximum number of matches per record in the input table to return. Must be between 0 and 10. 0 returns all matches.

threshold
number, The score threshold (between 0 and 1). Matches below this threshold will not be returned.

archived
boolean, Whether the Civis Data Match Job has been archived.

lastRun
list, A list containing the following elements:
• id integer,
• state string,
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.
• error string, The error message for this run, if present.

---

**enhancements_get_civis_data_match_runs**

*Check status of a run*

**Description**

Check status of a run

**Usage**

`enhancements_get_civis_data_match_runs(id, run_id)`
Arguments

- id: integer required. The ID of the civis_data_match.
- run_id: integer required. The ID of the run.

Value

A list containing the following elements:

- id: integer, The ID of the run.
- civisDataMatchId: integer, The ID of the civis_data_match.
- state: string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
- isCancelRequested: boolean, True if run cancel requested, else false.
- startedAt: string, The time the last run started at.
- finishedAt: string, The time the last run completed.
- error: string, The error, if any, returned by the run.

---

enforcements_get_geocode

*Get a Geocode Enhancement*

Description

Get a Geocode Enhancement

Usage

enforcements_get_geocode(id)

Arguments

- id: integer required.

Value

A list containing the following elements:

- id: integer, The ID for the enhancement.
- name: string, The name of the enhancement job.
- type: string, The type of the enhancement (e.g CASS-NCOA)
- createdAt: string, The time this enhancement was created.
- updatedAt: string, The time the enhancement was last updated.
- author: list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

state string, The status of the enhancement’s last run

schedule list, A list containing the following elements:
• scheduled boolean, If the item is scheduled.
• scheduledDays array, Day based on numeric value starting at 0 for Sunday.
• scheduledHours array, Hours of the day it is scheduled on.
• scheduledMinutes array, Minutes of the day it is scheduled on.
• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

parentId integer, Parent ID that triggers this enhancement.

notifications list, A list containing the following elements:
• urls array, URLs to receive a POST request at job completion
• successEmailSubject string, Custom subject line for success e-mail.
• successEmailBody string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

remoteHostId integer, The ID of the remote host.

credentialId integer, The ID of the remote host credential.

sourceSchemaAndTable string, The source database schema and table.

multipartKey array, The source table primary key.
limitingSQL string, The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').
targetSchema string, The output table schema.
targetTable string, The output table name.
country string, The country of the addresses to be geocoded; either 'us' or 'ca'.
provider string, The geocoding provider; one of postgis, nominatim, and geocoder_ca.
outputAddress boolean, Whether to output the parsed address. Only guaranteed for the 'postgis' provider.
archived string, The archival status of the requested item(s).

---

**enhancements_get_geocode_runs**

*Check status of a run*

**Description**

Check status of a run

**Usage**

enhancements_get_geocode_runs(id, run_id)

**Arguments**

id integer required. The ID of the geocode.
run_id integer required. The ID of the run.

**Value**

A list containing the following elements:

id integer, The ID of the run.
geocodeId integer, The ID of the geocode.
state string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested boolean, True if run cancel requested, else false.
startedAt string, The time the last run started at.
finishedAt string, The time the last run completed.
error string, The error, if any, returned by the run.
**enhancements_list**

*List Enhancements*

**Description**

List Enhancements

**Usage**

```python
enhancements_list(
    type = NULL,
    author = NULL,
    status = NULL,
    archived = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```

**Arguments**

- **type** string optional. If specified, return items of these types.
- **author** string optional. If specified, return items from this author. Must use user IDs. A comma separated list of IDs is also accepted to return items from multiple authors.
- **status** string optional. If specified, returns items with one of these statuses. It accepts a comma-separated list, possible values are 'running', 'failed', 'succeeded', 'idle', 'scheduled'.
- **archived** string optional. The archival status of the requested item(s).
- **limit** integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
- **page_num** integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order** string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at, last_run.updated_at.
- **order_dir** string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

**Value**

An array containing the following fields:

- **id** integer, The ID for the enhancement.
- **name** string, The name of the enhancement job.
enhancements_list_cass_ncoa_projects

List the projects a CASS/NCOA Enhancement belongs to

Description
List the projects a CASS/NCOA Enhancement belongs to

Usage
enhancements_list_cass_ncoa_projects(id, hidden = NULL)

Arguments
id integer required. The ID of the CASS/NCOA Enhancement.
hidden boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value
An array containing the following fields:
id integer, The ID for this project.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
name string, The name of this project.
description string, A description of the project.
users array, An array containing the following fields:
- id integer, The ID of this user.
- name string, This user’s name.
- username string, This user’s username.
- initials string, This user’s initials.
- online boolean, Whether this user is online.

autoShare boolean,
createdAt string,
updatedAt string,
archived string, The archival status of the requested item(s).

---

**enhancements_list_cass_ncoa_runs**

*List runs for the given cass_ncoa*

**Description**

List runs for the given cass_ncoa

**Usage**

```python
eenhancements_list_cass_ncoa_runs(
    id,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```

**Arguments**

- **id** integer required. The ID of the cass_ncoa.
- **limit** integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
- **page_num** integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order** string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
- **order_dir** string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.
Value

An array containing the following fields:

- **id** integer, The ID of the run.
- **cassNcoaId** integer, The ID of the cass_ncoa.
- **state** string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
- **isCancelRequested** boolean, True if run cancel requested, else false.
- **startedAt** string, The time the last run started at.
- **finishedAt** string, The time the last run completed.
- **error** string, The error, if any, returned by the run.

---

**enhancements_list_cass_ncoa_runs_logs**

*Get the logs for a run*

---

Description

Get the logs for a run

Usage

```
enhancements_list_cass_ncoa_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

- **id** integer required. The ID of the cass_ncoa.
- **run_id** integer required. The ID of the run.
- **last_id** integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted. Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
- **limit** integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

- **id** integer, The ID of the log.
- **createdAt** string, The time the log was created.
- **message** string, The log message.
- **level** string, The level of the log. One of unknown,fatal,error,warn,info,debug.
enhancements_list_cass_ncoa_runs_outputs

List the outputs for a run

Description

List the outputs for a run

Usage

enhancements_list_cass_ncoa_runs_outputs(
    id,
    run_id,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of the job.</td>
</tr>
<tr>
<td>run_id</td>
<td>integer required. The ID of the run.</td>
</tr>
<tr>
<td>limit</td>
<td>integer optional. Number of results to return. Defaults to its maximum of 50.</td>
</tr>
<tr>
<td>page_num</td>
<td>integer optional. Page number of the results to return. Defaults to the first page, 1.</td>
</tr>
<tr>
<td>order</td>
<td>string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.</td>
</tr>
<tr>
<td>order_dir</td>
<td>string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.</td>
</tr>
</tbody>
</table>

Value

An array containing the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>objectType</td>
<td>string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue</td>
</tr>
<tr>
<td>objectId</td>
<td>integer, The ID of the output.</td>
</tr>
<tr>
<td>name</td>
<td>string, The name of the output.</td>
</tr>
<tr>
<td>link</td>
<td>string, The hypermedia link to the output.</td>
</tr>
<tr>
<td>value</td>
<td>string,</td>
</tr>
</tbody>
</table>
**Description**

List users and groups permissioned on this object

**Usage**

`enhancements_list_cass_ncoa_shares(id)`

**Arguments**

- **id** integer required. The ID of the resource that is shared.

**Value**

An array containing the following fields:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,
- **writers** list, A list containing the following elements:
  - users array,
  - groups array,
- **owners** list, A list containing the following elements:
  - users array,
  - groups array,
- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.
Enhancements List Civis Data Match Projects

List the projects a Civis Data Match Enhancement belongs to

Description

List the projects a Civis Data Match Enhancement belongs to

Usage

```
enhancements_list_civis_data_match_projects(id, hidden = NULL)
```

Arguments

- **id**: integer required. The ID of the Civis Data Match Enhancement.
- **hidden**: boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

- **id**: integer, The ID for this project.
- **author**: list, A list containing the following elements:
  - **id**: integer, The ID of this user.
  - **name**: string, This user’s name.
  - **username**: string, This user’s username.
  - **initials**: string, This user’s initials.
  - **online**: boolean, Whether this user is online.
- **name**: string, The name of this project.
- **description**: string, A description of the project.
- **users**: array, An array containing the following fields:
  - **id**: integer, The ID of this user.
  - **name**: string, This user’s name.
  - **username**: string, This user’s username.
  - **initials**: string, This user’s initials.
  - **online**: boolean, Whether this user is online.
- **autoShare**: boolean,
- **createdAt**: string,
- **updatedAt**: string,
- **archived**: string, The archival status of the requested item(s).
enhancements_list_civis_data_match_runs

List runs for the given civis_data_match

Description

List runs for the given civis_data_match

Usage

enhancements_list_civis_data_match_runs(
    id,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)

Arguments

id integer required. The ID of the civis_data_match.

limit integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.

page_num integer optional. Page number of the results to return. Defaults to the first page, 1.

order string optional. The field on which to order the result set. Defaults to id. Must be one of: id.

order_dir string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id integer, The ID of the run.
civisDataMatchId integer, The ID of the civis_data_match.
state string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested boolean, True if run cancel requested, else false.
startedAt string, The time the last run started at.
finishedAt string, The time the last run completed.
error string, The error, if any, returned by the run.
enhancements_list_civis_data_match_runs_logs

*Get the logs for a run*

**Description**

Get the logs for a run

**Usage**

```python
enhancements_list_civis_data_match_runs_logs(
    id,
    run_id,
    last_id = NULL,
    limit = NULL
)
```

**Arguments**

- **id**
  - integer required. The ID of the civis_data_match.
- **run_id**
  - integer required. The ID of the run.
- **last_id**
  - integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted. Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
- **limit**
  - integer optional. The maximum number of log messages to return. Default of 10000.

**Value**

An array containing the following fields:

- **id**
  - integer, The ID of the log.
- **createdAt**
  - string, The time the log was created.
- **message**
  - string, The log message.
- **level**
  - string, The level of the log. One of unknown,fatal,error,warn,info,debug.
enhancements_list_civis_data_match_runs_outputs

List the outputs for a run

**Description**

List the outputs for a run

**Usage**

```r
enhancements_list_civis_data_match_runs_outputs(
  id,  
  run_id,  
  limit = NULL,  
  page_num = NULL,  
  order = NULL,  
  order_dir = NULL
)
```

**Arguments**

- **id**
  - integer required. The ID of the job.
- **run_id**
  - integer required. The ID of the run.
- **limit**
  - integer optional. Number of results to return. Defaults to its maximum of 50.
- **page_num**
  - integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order**
  - string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
- **order_dir**
  - string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

**Value**

An array containing the following fields:

- **objectType**
  - string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
- **objectId**
  - integer, The ID of the output.
- **name**
  - string, The name of the output.
- **link**
  - string, The hypermedia link to the output.
- **value**
  - string,
Description

List users and groups permissioned on this object

Usage

enhancements_list_civis_data_match_shares(id)

Arguments

id integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,

- **writers** list, A list containing the following elements:
  - users array,
  - groups array,

- **owners** list, A list containing the following elements:
  - users array,
  - groups array,

- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.
**enhancements_list_field_mapping**

List the fields in a field mapping for Civis Data Match, Data Unification, and Table Deduplication jobs

**Description**

List the fields in a field mapping for Civis Data Match, Data Unification, and Table Deduplication jobs

**Usage**

enhancements_list_field_mapping()

**Value**

An array containing the following fields:

- **field**: string, The name of the field.
- **description**: string, The description of the field.

**enhancements_list_geocode_projects**

List the projects a Geocode Enhancement belongs to

**Description**

List the projects a Geocode Enhancement belongs to

**Usage**

enhancements_list_geocode_projects(id, hidden = NULL)

**Arguments**

- **id**: integer required. The ID of the Geocode Enhancement.
- **hidden**: boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
Value

An array containing the following fields:

- **id** integer, The ID for this project.
- **author** list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.
- **name** string, The name of this project.
- **description** string, A description of the project.
- **users** array, An array containing the following fields:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.
- **autoShare** boolean,
- **createdAt** string,
- **updatedAt** string,
- **archived** string, The archival status of the requested item(s).

```
enhancements_list_geocode_runs

List runs for the given geocode
```

Description

List runs for the given geocode

Usage

```
enhancements_list_geocode_runs(
    id,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td>required. The ID of the geocode.</td>
</tr>
<tr>
<td>limit</td>
<td>integer</td>
<td>optional. Number of results to return. Defaults to 20. Maximum allowed is 100.</td>
</tr>
<tr>
<td>page_num</td>
<td>integer</td>
<td>optional. Page number of the results to return. Defaults to the first page, 1.</td>
</tr>
<tr>
<td>order</td>
<td>string</td>
<td>optional. The field on which to order the result set. Defaults to id. Must be one of: id.</td>
</tr>
<tr>
<td>order_dir</td>
<td>string</td>
<td>optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.</td>
</tr>
</tbody>
</table>

Value

An array containing the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td>integer, The ID of the run.</td>
</tr>
<tr>
<td>geocodeId</td>
<td>integer</td>
<td>integer, The ID of the geocode.</td>
</tr>
<tr>
<td>state</td>
<td>string</td>
<td>string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.</td>
</tr>
<tr>
<td>isCancelRequested</td>
<td>boolean</td>
<td>boolean, True if run cancel requested, else false.</td>
</tr>
<tr>
<td>startedAt</td>
<td>string</td>
<td>string, The time the last run started at.</td>
</tr>
<tr>
<td>finishedAt</td>
<td>string</td>
<td>string, The time the last run completed.</td>
</tr>
<tr>
<td>error</td>
<td>string</td>
<td>string, The error, if any, returned by the run.</td>
</tr>
</tbody>
</table>

enhancements_list_geocode_runs_logs

Get the logs for a run

Description

Get the logs for a run

Usage

enhancements_list_geocode_runs_logs(id, run_id, last_id = NULL, limit = NULL)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td>integer required. The ID of the geocode.</td>
</tr>
<tr>
<td>run_id</td>
<td>integer</td>
<td>integer required. The ID of the run.</td>
</tr>
<tr>
<td>last_id</td>
<td>integer</td>
<td>integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted. Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.</td>
</tr>
<tr>
<td>limit</td>
<td>integer</td>
<td>integer optional. The maximum number of log messages to return. Default of 10000.</td>
</tr>
</tbody>
</table>
Value

An array containing the following fields:

- **id**    integer, The ID of the log.
- **createdAt**    string, The time the log was created.
- **message**    string, The log message.
- **level**    string, The level of the log. One of unknown,fatal,error,warn,info,debug.

```

discoveries_list_geocode_runs_outputs

List the outputs for a run

```

Description

List the outputs for a run

Usage

```

discoveries_list_geocode_runs_outputs(
    id,
    run_id,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)

```

Arguments

- **id**    integer required. The ID of the job.
- **run_id**    integer required. The ID of the run.
- **limit**    integer optional. Number of results to return. Defaults to its maximum of 50.
- **page_num**    integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order**    string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
- **order_dir**    string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.
Value

An array containing the following fields:

- **objectType** string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
- **objectId** integer, The ID of the output.
- **name** string, The name of the output.
- **link** string, The hypermedia link to the output.
- **value** string,

---

**enhancements_list_geocode_shares**

*List users and groups permissioned on this object*

---

Description

List users and groups permissioned on this object

Usage

```
enhancements_list_geocode_shares(id)
```

Arguments

- **id** integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,
- **writers** list, A list containing the following elements:
  - users array,
  - groups array,
- **owners** list, A list containing the following elements:
  - users array,
  - groups array,
- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.
**enhancements_list_types**

*List available enhancement types*

**Description**

List available enhancement types

**Usage**

`enhancements_list_types()`

**Value**

An array containing the following fields:

- `name` string, The name of the type.

**enhancements_patch_cass_ncoa**

*Update some attributes of this CASS/NCOA Enhancement*

**Description**

Update some attributes of this CASS/NCOA Enhancement

**Usage**

```python
enhancements_patch_cass_ncoa(
    id,
    name = NULL,
    schedule = NULL,
    parent_id = NULL,
    notifications = NULL,
    source = NULL,
    destination = NULL,
    column_mapping = NULL,
    use_default_column_mapping = NULL,
    perform_ncoa = NULL,
    ncoa_credential_id = NULL,
    output_level = NULL,
    limiting_sql = NULL
)
```
Arguments

id integer required. The ID for the enhancement.
name string optional. The name of the enhancement job.
schedule list optional. A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.
parent_id integer optional. Parent ID that triggers this enhancement.
notifications list optional. A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  • successOn boolean, If success email notifications are on.
  • failureOn boolean, If failure email notifications are on.
source list optional. A list containing the following elements:
  • databaseTable list . A list containing the following elements:
    – schema string, The schema name of the source table.
    – table string, The name of the source table.
    – remoteHostId integer, The ID of the database host for the table.
    – credentialId integer, The id of the credentials to be used when performing the enhancement.
    – multipartKey array, The source table primary key.
destination list optional. A list containing the following elements:
  • databaseTable list . A list containing the following elements:
    – schema string, The schema name for the output data.
    – table string, The table name for the output data.
column_mapping list optional. A list containing the following elements:
• address1 string, The first address line.
• address2 string, The second address line.
• city string, The city of an address.
• state string, The state of an address.
• zip string, The zip code of an address.
• name string, The full name of the resident at this address. If needed, separate multiple columns with ‘+’, e.g. ‘first_name+last_name’
• company string, The name of the company located at this address.

use_default_column_mapping
boolean optional. Defaults to true, where the existing column mapping on the input table will be used. If false, a custom column mapping must be provided.

perform_ncoa
boolean optional. Whether to update addresses for records matching the National Change of Address (NCOA) database.

ncoa_credential_id
integer optional. Credential to use when performing NCOA updates. Required if ‘performNcoa’ is true.

output_level
string optional. The set of fields persisted by a CASS or NCOA enhancement. For CASS enhancements, one of 'cass' or 'all.' For NCOA enhancements, one of 'cass', 'ncoa', 'coalesced' or 'all'. By default, all fields will be returned.

limiting_sql
string optional. The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').

Value

A list containing the following elements:

id integer, The ID for the enhancement.
name string, The name of the enhancement job.
type string, The type of the enhancement (e.g CASS-NCOA)
createdAt string, The time this enhancement was created.
updatedAt string, The time the enhancement was last updated.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
state string, The status of the enhancement’s last run
schedule list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

parentId integer, Parent ID that triggers this enhancement.

notifications list, A list containing the following elements:
  - urls array, URLs to receive a POST request at job completion
  - successEmailSubject string, Custom subject line for success e-mail.
  - successEmailBody string, Custom body text for success e-mail, written in Markdown.
  - successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  - successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  - successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  - failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  - stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  - successOn boolean, If success email notifications are on.
  - failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.

source list, A list containing the following elements:
  - databaseTable list. A list containing the following elements:
    - schema string, The schema name of the source table.
    - table string, The name of the source table.
    - remoteHostId integer, The ID of the database host for the table.
    - credentialId integer, The id of the credentials to be used when performing the enhancement.
    - multipartKey array, The source table primary key.

destination list, A list containing the following elements:
  - databaseTable list. A list containing the following elements:
    - schema string, The schema name for the output data.
    - table string, The table name for the output data.

columnMapping list, A list containing the following elements:
  - address1 string, The first address line.
  - address2 string, The second address line.
  - city string, The city of an address.
 enhancements_patch_civis_data_match

 - state string, The state of an address.
 - zip string, The zip code of an address.
 - name string, The full name of the resident at this address. If needed, separate multiple columns with ‘+’, e.g. ‘first_name+last_name’
 - company string, The name of the company located at this address.

useDefaultColumnMapping

boolean, Defaults to true, where the existing column mapping on the input table will be used. If false, a custom column mapping must be provided.

performNcoa

boolean, Whether to update addresses for records matching the National Change of Address (NCOA) database.

ncoaCredentialId

integer, Credential to use when performing NCOA updates. Required if 'performNcoa' is true.

outputLevel

string, The set of fields persisted by a CASS or NCOA enhancement. For CASS enhancements, one of 'cass' or 'all'. For NCOA enhancements, one of 'cass', 'ncoa', 'coalesced' or 'all'. By default, all fields will be returned.

limitingSQL

string, The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').

archived

string, The archival status of the requested item(s).

Description

Update some attributes of this Civis Data Match Enhancement

Usage

```javascript
enhancements_patch_civis_data_match(
  id,
  name = NULL,
  schedule = NULL,
  parent_id = NULL,
  notifications = NULL,
  input_field_mapping = NULL,
  input_table = NULL,
  match_target_id = NULL,
  output_table = NULL,
  max_matches = NULL,
  threshold = NULL,
  archived = NULL
)
```
Arguments

**id**
integer required. The ID for the enhancement.

**name**
string optional. The name of the enhancement job.

**schedule**
list optional. A list containing the following elements:
- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

**parent_id**
integer optional. Parent ID that triggers this enhancement.

**notifications**
list optional. A list containing the following elements:
- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

**input_field_mapping**
list optional. The column mapping for the input table. See /enhancements/field_mapping for list of valid fields.

**input_table**
list optional. A list containing the following elements:
- databaseName string, The Redshift database name for the table.
- schema string, The schema name for the table.
- table string, The table name.

**match_target_id**
integer optional. The ID of the Civis Data match target. See /match_targets for IDs.

**output_table**
list optional. A list containing the following elements:
- databaseName string, The Redshift database name for the table.
- schema string, The schema name for the table.
- table string, The table name.
max_matches: integer optional. The maximum number of matches per record in the input table to return. Must be between 0 and 10. 0 returns all matches.

threshold: number optional. The score threshold (between 0 and 1). Matches below this threshold will not be returned.

archived: boolean optional. Whether the Civis Data Match Job has been archived.

Value

A list containing the following elements:

id: integer, The ID for the enhancement.

name: string, The name of the enhancement job.

type: string, The type of the enhancement (e.g CASS-NCOA)

createdAt: string, The time this enhancement was created.

updatedAt: string, The time the enhancement was last updated.

author: list, A list containing the following elements:

  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

state: string, The status of the enhancement’s last run

schedule: list, A list containing the following elements:

  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

parentId: integer, Parent ID that triggers this enhancement.

notifications: list, A list containing the following elements:

  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

inputFieldMapping list, The column mapping for the input table. See /enhancements/field_mapping for list of valid fields.

inputTable list, A list containing the following elements:
• databaseName string, The Redshift database name for the table.
• schema string, The schema name for the table.
• table string, The table name.

matchTargetId integer, The ID of the Civis Data match target. See /match_targets for IDs.

outputTable list, A list containing the following elements:
• databaseName string, The Redshift database name for the table.
• schema string, The schema name for the table.
• table string, The table name.

maxMatches integer, The maximum number of matches per record in the input table to return. Must be between 0 and 10. 0 returns all matches.

threshold number, The score threshold (between 0 and 1). Matches below this threshold will not be returned.

archived boolean, Whether the Civis Data Match Job has been archived.

lastRun list, A list containing the following elements:
• id integer,
• state string,
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.
• error string, The error message for this run, if present.
Update some attributes of this Geocode Enhancement

Usage

```
enhancements_patch_geocode(
    id,
    name = NULL,
    schedule = NULL,
    parent_id = NULL,
    notifications = NULL,
    remote_host_id = NULL,
    credential_id = NULL,
    source_schema_and_table = NULL,
    multipart_key = NULL,
    limiting_sql = NULL,
    target_schema = NULL,
    target_table = NULL,
    country = NULL,
    provider = NULL,
    output_address = NULL
)
```

Arguments

**id**
integer required. The ID for the enhancement.

**name**
string optional. The name of the enhancement job.

**schedule**
list optional. A list containing the following elements:
- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

**parent_id**
integer optional. Parent ID that triggers this enhancement.

**notifications**
list optional. A list containing the following elements:
- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

remote_host_id integer optional. The ID of the remote host.
credential_id integer optional. The ID of the remote host credential.
source_schema_and_table string optional. The source database schema and table.
multipart_key array optional. The source table primary key.
limiting_sql string optional. The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').
target_schema string optional. The output table schema.
target_table string optional. The output table name.
country string optional. The country of the addresses to be geocoded; either 'us' or 'ca'.
provider string optional. The geocoding provider; one of postgis, nominatim, and geocoder_ca.
output_address boolean optional. Whether to output the parsed address. Only guaranteed for the 'postgis' provider.

Value

A list containing the following elements:

- id integer, The ID for the enhancement.
- name string, The name of the enhancement job.
- type string, The type of the enhancement (e.g CASS-NCOA)
- createdAt string, The time this enhancement was created.
- updatedAt string, The time the enhancement was last updated.
- author list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>state</td>
<td>string, The status of the enhancement’s last run</td>
</tr>
<tr>
<td>schedule</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• scheduled boolean, If the item is scheduled.</td>
</tr>
<tr>
<td></td>
<td>• scheduledDays array, Day based on numeric value starting at 0 for Sunday.</td>
</tr>
<tr>
<td></td>
<td>• scheduledHours array, Hours of the day it is scheduled on.</td>
</tr>
<tr>
<td></td>
<td>• scheduledMinutes array, Minutes of the day it is scheduled on.</td>
</tr>
<tr>
<td></td>
<td>• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.</td>
</tr>
<tr>
<td>parentId</td>
<td>integer, Parent ID that triggers this enhancement.</td>
</tr>
<tr>
<td>notifications</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• urls array, URLs to receive a POST request at job completion</td>
</tr>
<tr>
<td></td>
<td>• successEmailSubject string, Custom subject line for success e-mail.</td>
</tr>
<tr>
<td></td>
<td>• successEmailBody string, Custom body text for success e-mail, written in Markdown.</td>
</tr>
<tr>
<td></td>
<td>• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.</td>
</tr>
<tr>
<td></td>
<td>• successEmailFromName string, Name from which success emails are sent; defaults to &quot;Civis.&quot;</td>
</tr>
<tr>
<td></td>
<td>• successEmailReplyTo string, Address for replies to sucess emails; defaults to the author of the job.</td>
</tr>
<tr>
<td></td>
<td>• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.</td>
</tr>
<tr>
<td></td>
<td>• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.</td>
</tr>
<tr>
<td></td>
<td>• successOn boolean, If success email notifications are on.</td>
</tr>
<tr>
<td></td>
<td>• failureOn boolean, If failure email notifications are on.</td>
</tr>
<tr>
<td>runningAs</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• id integer, The ID of this user.</td>
</tr>
<tr>
<td></td>
<td>• name string, This user’s name.</td>
</tr>
<tr>
<td></td>
<td>• username string, This user’s username.</td>
</tr>
<tr>
<td></td>
<td>• initials string, This user’s initials.</td>
</tr>
<tr>
<td></td>
<td>• online boolean, Whether this user is online.</td>
</tr>
<tr>
<td>remoteHostId</td>
<td>integer, The ID of the remote host.</td>
</tr>
<tr>
<td>credentialId</td>
<td>integer, The ID of the remote host credential.</td>
</tr>
<tr>
<td>sourceSchemaAndTable</td>
<td>string, The source database schema and table.</td>
</tr>
<tr>
<td>multipartKey</td>
<td>array, The source table primary key.</td>
</tr>
<tr>
<td>limitingSQL</td>
<td>string, The limiting SQL for the source table. &quot;WHERE&quot; should be omitted (e.g. state='IL').</td>
</tr>
<tr>
<td>targetSchema</td>
<td>string, The output table schema.</td>
</tr>
</tbody>
</table>
targetTable string, The output table name.
country string, The country of the addresses to be geocoded; either 'us' or 'ca'.
provider string, The geocoding provider; one of postgis, nominatim, and geocoder_ca.
outputAddress bool, Whether to output the parsed address. Only guaranteed for the 'postgis' provider.
archived string, The archival status of the requested item(s).

enhancements_post_cass_ncoa

Create a CASS/NCOA Enhancement

Description
Create a CASS/NCOA Enhancement

Usage

enhancements_post_cass_ncoa(
  name,
  source,
  schedule = NULL,
  parent_id = NULL,
  notifications = NULL,
  destination = NULL,
  column_mapping = NULL,
  use_default_column_mapping = NULL,
  perform_ncoa = NULL,
  ncoa_credential_id = NULL,
  output_level = NULL,
  limiting_sql = NULL
)

Arguments

name string required. The name of the enhancement job.
source list required. A list containing the following elements:
  • databaseTable list. A list containing the following elements:
    – schema string, The schema name of the source table.
    – table string, The name of the source table.
    – remoteHostId integer, The ID of the database host for the table.
    – credentialId integer, The id of the credentials to be used when performing the enhancement.
    – multipartKey array, The source table primary key.
schedule list optional. A list containing the following elements:
• scheduled boolean, If the item is scheduled.
• scheduledDays array, Day based on numeric value starting at 0 for Sunday.
• scheduledHours array, Hours of the day it is scheduled on.
• scheduledMinutes array, Minutes of the day it is scheduled on.
• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

parent_id integer optional. Parent ID that triggers this enhancement.

notifications list optional. A list containing the following elements:
• urls array, URLs to receive a POST request at job completion
• successEmailSubject string, Custom subject line for success e-mail.
• successEmailBody string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

destination list optional. A list containing the following elements:
• databaseTable list . A list containing the following elements:
  – schema string, The schema name for the output data.
  – table string, The table name for the output data.

column_mapping list optional. A list containing the following elements:
• address1 string, The first address line.
• address2 string, The second address line.
• city string, The city of an address.
• state string, The state of an address.
• zip string, The zip code of an address.
• name string, The full name of the resident at this address. If needed, separate multiple columns with ‘+’, e.g. 'first_name+last_name'
• company string, The name of the company located at this address.

use_default_column_mapping boolean optional. Defaults to true, where the existing column mapping on the input table will be used. If false, a custom column mapping must be provided.

perform_ncoa boolean optional. Whether to update addresses for records matching the National Change of Address (NCOA) database.
ncoa_credential_id
integer optional. Credential to use when performing NCOA updates. Required if 'performNcoa' is true.

output_level
string optional. The set of fields persisted by a CASS or NCOA enhancement. For CASS enhancements, one of 'cass' or 'all'. For NCOA enhancements, one of 'cass', 'ncoa', 'coalesced' or 'all'. By default, all fields will be returned.

limiting_sql
string optional. The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').

Value
A list containing the following elements:

id
integer, The ID for the enhancement.

name
string, The name of the enhancement job.

type
string, The type of the enhancement (e.g. CASS-NCOA).

createdAt
string, The time this enhancement was created.

updatedAt
string, The time the enhancement was last updated.

author
list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user's name.
  • username string, This user's username.
  • initials string, This user's initials.
  • online boolean, Whether this user is online.

state
string, The status of the enhancement's last run.

schedule
list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

parentId
integer, Parent ID that triggers this enhancement.

notifications
list, A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion.
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user's name.
• username string, This user's username.
• initials string, This user's initials.
• online boolean, Whether this user is online.

source list, A list containing the following elements:
• databaseTable list, A list containing the following elements:
  – schema string, The schema name of the source table.
  – table string, The name of the source table.
  – remoteHostId integer, The ID of the database host for the table.
  – credentialId integer, The id of the credentials to be used when performing the enhancement.
  – multipartKey array, The source table primary key.

destination list, A list containing the following elements:
• databaseTable list, A list containing the following elements:
  – schema string, The schema name for the output data.
  – table string, The table name for the output data.

columnMapping list, A list containing the following elements:
• address1 string, The first address line.
• address2 string, The second address line.
• city string, The city of an address.
• state string, The state of an address.
• zip string, The zip code of an address.
• name string, The full name of the resident at this address. If needed, separate multiple columns with '+', e.g. 'first_name+last_name'
• company string, The name of the company located at this address.

useDefaultColumnMapping boolean, Defaults to true, where the existing column mapping on the input table will be used. If false, a custom column mapping must be provided.

performNcoa boolean, Whether to update addresses for records matching the National Change of Address (NCOA) database.

ncoaCredentialId integer, Credential to use when performing NCOA updates. Required if 'performNcoa' is true.
outputLevel  string, The set of fields persisted by a CASS or NCOA enhancement. For CASS enhancements, one of 'cass' or 'all'. For NCOA enhancements, one of 'cass', 'ncoa', 'coalesced' or 'all'. By default, all fields will be returned.

limitingSQL  string, The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').

archived  string, The archival status of the requested item(s).

---

**enhancements_post_cass_ncoa_cancel**

*Cancel a run*

**Description**

Cancel a run

**Usage**

enhancements_post_cass_ncoa_cancel(id)

**Arguments**

id  integer required. The ID of the job.

**Value**

A list containing the following elements:

id  integer, The ID of the run.

state  string, The state of the run, one of 'queued', 'running' or 'cancelled'.

isCancelRequested  boolean, True if run cancel requested, else false.

---

**enhancements_post_cass_ncoa_runs**

*Start a run*

**Description**

Start a run

**Usage**

enhancements_post_cass_ncoa_runs(id)
**Enhancements Post Civis Data Match**

**Arguments**

| id | integer required. The ID of the `cass_ncoa`. |

**Value**

A list containing the following elements:

| id      | integer, The ID of the run. |
| cassNcoaId | integer, The ID of the `cass_ncoa`. |
| isCancelRequested | boolean, True if run cancel requested, else false. |
| startedAt | string, The time the last run started at. |
| finishedAt | string, The time the last run completed. |
| error   | string, The error, if any, returned by the run. |

---

**Description**

Create a Civis Data Match Enhancement

**Usage**

```python
enhancements_post_civis_data_match(
    name,
    input_field_mapping,
    input_table,
    match_target_id,
    output_table,
    schedule = NULL,
    parent_id = NULL,
    notifications = NULL,
    max_matches = NULL,
    threshold = NULL,
    archived = NULL
)
```
Arguments

**name**
- string required. The name of the enhancement job.

**input_field_mapping**
- list required. The column mapping for the input table. See `/enhancements/field_mapping` for list of valid fields.

**input_table**
- list required. A list containing the following elements:
  - databaseName string, The Redshift database name for the table.
  - schema string, The schema name for the table.
  - table string, The table name.

**match_target_id**
- integer required. The ID of the Civis Data match target. See `/match_targets` for IDs.

**output_table**
- list required. A list containing the following elements:
  - databaseName string, The Redshift database name for the table.
  - schema string, The schema name for the table.
  - table string, The table name.

**schedule**
- list optional. A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

**parent_id**
- integer optional. Parent ID that triggers this enhancement.

**notifications**
- list optional. A list containing the following elements:
  - urls array, URLs to receive a POST request at job completion
  - successEmailSubject string, Custom subject line for success e-mail.
  - successEmailBody string, Custom body text for success e-mail, written in Markdown.
  - successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  - successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  - successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  - failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  - stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  - successOn boolean, If success email notifications are on.
  - failureOn boolean, If failure email notifications are on.
**max_matches** integer optional. The maximum number of matches per record in the input table to return. Must be between 0 and 10. 0 returns all matches.

**threshold** number optional. The score threshold (between 0 and 1). Matches below this threshold will not be returned.

**archived** boolean optional. Whether the Civis Data Match Job has been archived.

**Value**

A list containing the following elements:

- **id** integer, The ID for the enhancement.
- **name** string, The name of the enhancement job.
- **type** string, The type of the enhancement (e.g. CASS-NCOA)
- **createdAt** string, The time this enhancement was created.
- **updatedAt** string, The time the enhancement was last updated.
- **author** list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.
- **state** string, The status of the enhancement’s last run
- **schedule** list, A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.
- **parentId** integer, Parent ID that triggers this enhancement.
- **notifications** list, A list containing the following elements:
  - urls array, URLs to receive a POST request at job completion
  - successEmailSubject string, Custom subject line for success e-mail.
  - successEmailBody string, Custom body text for success e-mail, written in Markdown.
  - successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  - successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  - successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  - failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user's name.
• username string, This user's username.
• initials string, This user's initials.
• online boolean, Whether this user is online.

inputFieldMapping list, The column mapping for the input table. See /enhancements/field_mapping for list of valid fields.

inputTable list, A list containing the following elements:
• databaseName string, The Redshift database name for the table.
• schema string, The schema name for the table.
• table string, The table name.

matchTargetId integer, The ID of the Civis Data match target. See /match_targets for IDs.

outputTable list, A list containing the following elements:
• databaseName string, The Redshift database name for the table.
• schema string, The schema name for the table.
• table string, The table name.

maxMatches integer, The maximum number of matches per record in the input table to return. Must be between 0 and 10. 0 returns all matches.

threshold number, The score threshold (between 0 and 1). Matches below this threshold will not be returned.

archived boolean, Whether the Civis Data Match Job has been archived.

lastRun list, A list containing the following elements:
• id integer,
• state string,
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.
• error string, The error message for this run, if present.
**enhancements_post_civis_data_match_cancel**

*Cancel a run*

**Description**

Cancel a run

**Usage**

`enhancements_post_civis_data_match_cancel(id)`

**Arguments**

- `id` integer required. The ID of the job.

**Value**

A list containing the following elements:

- `id` integer, The ID of the run.
- `state` string, The state of the run, one of 'queued', 'running' or 'cancelled'.
- `isCancelRequested` boolean, True if run cancel requested, else false.

---

**enhancements_post_civis_data_match_clone**

*Clone this Civis Data Match Enhancement*

**Description**

Clone this Civis Data Match Enhancement

**Usage**

```python
enhancements_post_civis_data_match_clone(
    id,
    clone_schedule = NULL,
    clone_triggers = NULL,
    clone_notifications = NULL
)
```
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID for the enhancement.</td>
</tr>
<tr>
<td>clone_schedule</td>
<td>boolean optional. If true, also copy the schedule to the new enhancement.</td>
</tr>
<tr>
<td>clone_triggers</td>
<td>boolean optional. If true, also copy the triggers to the new enhancement.</td>
</tr>
<tr>
<td>clone_notifications</td>
<td>boolean optional. If true, also copy the notifications to the new enhancement.</td>
</tr>
</tbody>
</table>

Value

A list containing the following elements:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer, The ID for the enhancement.</td>
</tr>
<tr>
<td>name</td>
<td>string, The name of the enhancement job.</td>
</tr>
<tr>
<td>type</td>
<td>string, The type of the enhancement (e.g., CASS-NCOA).</td>
</tr>
<tr>
<td>createdAt</td>
<td>string, The time this enhancement was created.</td>
</tr>
<tr>
<td>updatedAt</td>
<td>string, The time the enhancement was last updated.</td>
</tr>
<tr>
<td>author</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td>state</td>
<td>string, The status of the enhancement’s last run</td>
</tr>
<tr>
<td>schedule</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td>parentId</td>
<td>integer, Parent ID that triggers this enhancement.</td>
</tr>
<tr>
<td>notifications</td>
<td>list, A list containing the following elements:</td>
</tr>
</tbody>
</table>

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user's name.
• username string, This user's username.
• initials string, This user's initials.
• online boolean, Whether this user is online.

inputFieldMapping list, The column mapping for the input table. See /enhancements/field_mapping for list of valid fields.

inputTable list, A list containing the following elements:
• databaseName string, The Redshift database name for the table.
• schema string, The schema name for the table.
• table string, The table name.

matchTargetId integer, The ID of the Civis Data match target. See /match_targets for IDs.

outputTable list, A list containing the following elements:
• databaseName string, The Redshift database name for the table.
• schema string, The schema name for the table.
• table string, The table name.

maxMatches integer, The maximum number of matches per record in the input table to return. Must be between 0 and 10. 0 returns all matches.

threshold number, The score threshold (between 0 and 1). Matches below this threshold will not be returned.

archived boolean, Whether the Civis Data Match Job has been archived.

lastRun list, A list containing the following elements:
• id integer,
• state string,
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.
• error string, The error message for this run, if present.
**enhancements_post_geocode**

Create a Geocode Enhancement

**Description**

Create a Geocode Enhancement

**enhancements_post_civis_data_match_runs**

Start a run

**Description**

Start a run

**Usage**

`enhancements_post_civis_data_match_runs(id)`

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of the civis_data_match.</td>
</tr>
</tbody>
</table>

**Value**

A list containing the following elements:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer, The ID of the run.</td>
</tr>
<tr>
<td>civisDataMatchId</td>
<td>integer, The ID of the civis_data_match.</td>
</tr>
<tr>
<td>state</td>
<td>string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.</td>
</tr>
<tr>
<td>isCancelRequested</td>
<td>boolean, True if run cancel requested, else false.</td>
</tr>
<tr>
<td>startedAt</td>
<td>string, The time the last run started at.</td>
</tr>
<tr>
<td>finishedAt</td>
<td>string, The time the last run completed.</td>
</tr>
<tr>
<td>error</td>
<td>string, The error, if any, returned by the run.</td>
</tr>
</tbody>
</table>
enhancements_post_geocode

Usage

eenhancements_post_geocode(
    name,
    remote_host_id,
    credential_id,
    source_schema_and_table,
    schedule = NULL,
    parent_id = NULL,
    notifications = NULL,
    multipart_key = NULL,
    limiting_sql = NULL,
    target_schema = NULL,
    target_table = NULL,
    country = NULL,
    provider = NULL,
    output_address = NULL
)

Arguments

name string required. The name of the enhancement job.
remote_host_id integer required. The ID of the remote host.
credential_id integer required. The ID of the remote host credential.
schedule string required. The source database schema and table.
schedule list optional. A list containing the following elements:
    • scheduled boolean, If the item is scheduled.
    • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
    • scheduledHours array, Hours of the day it is scheduled on.
    • scheduledMinutes array, Minutes of the day it is scheduled on.
    • scheduledRunsPerHour integer, Alternative to scheduled minutes, number
      of times to run per hour.
parent_id integer optional. Parent ID that triggers this enhancement.
notifications list optional. A list containing the following elements:
    • urls array, URLs to receive a POST request at job completion
    • successEmailSubject string, Custom subject line for success e-mail.
    • successEmailBody string, Custom body text for success e-mail, written in
      Markdown.
    • successEmailAddresses array, Addresses to notify by e-mail when the job
      completes successfully.
    • successEmailFromName string, Name from which success emails are sent;
      defaults to "Civis."
    • successEmailReplyTo string, Address for replies to success emails; defaults
      to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

multipart_key array optional. The source table primary key.
limiting_sql string optional. The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').
target_schema string optional. The output table schema.
target_table string optional. The output table name.
country string optional. The country of the addresses to be geocoded; either 'us' or 'ca'.
provider string optional. The geocoding provider; one of postgis, nominatim, and geocoder_ca.
output_address boolean optional. Whether to output the parsed address. Only guaranteed for the 'postgis' provider.

Value

A list containing the following elements:

- id integer, The ID for the enhancement.
- name string, The name of the enhancement job.
- type string, The type of the enhancement (e.g CASS-NCOA)
- createdAt string, The time this enhancement was created.
- updatedAt string, The time the enhancement was last updated.
- author list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user's name.
  - username string, This user's username.
  - initials string, This user's initials.
  - online boolean, Whether this user is online.
- state string, The status of the enhancement's last run
- schedule list, A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.
- parentId integer, Parent ID that triggers this enhancement.
- notifications list, A list containing the following elements:
• urls array, URLs to receive a POST request at job completion
• successEmailSubject string, Custom subject line for success e-mail.
• successEmailBody string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

remoteHostId integer, The ID of the remote host.
credentialId integer, The ID of the remote host credential.
sourceSchemaAndTable string, The source database schema and table.
multipartKey array, The source table primary key.
limitingSQL string, The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').
targetSchema string, The output table schema.
targetTable string, The output table name.
country string, The country of the addresses to be geocoded; either 'us' or 'ca'.
provider string, The geocoding provider; one of postgis, nominatim, and geocoder_ca.
outputAddress boolean, Whether to output the parsed address. Only guaranteed for the 'postgis' provider.
archived string, The archival status of the requested item(s).
**enhancements_post_geocode_cancel**

*Cancel a run*

**Description**

Cancel a run

**Usage**

`enhancements_post_geocode_cancel(id)`

**Arguments**

- **id**
  
  integer required. The ID of the job.

**Value**

A list containing the following elements:

- **id**
  
  integer, The ID of the run.

- **state**
  
  string, The state of the run, one of 'queued', 'running' or 'cancelled'.

- **isCancelRequested**
  
  boolean, True if run cancel requested, else false.

---

**enhancements_post_geocode_runs**

*Start a run*

**Description**

Start a run

**Usage**

`enhancements_post_geocode_runs(id)`

**Arguments**

- **id**
  
  integer required. The ID of the geocode.
Value

A list containing the following elements:

- **id**: integer, The ID of the run.
- **geocodeId**: integer, The ID of the geocode.
- **state**: string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
- **isCancelRequested**: boolean, True if run cancel requested, else false.
- **startedAt**: string, The time the last run started at.
- **finishedAt**: string, The time the last run completed.
- **error**: string, The error, if any, returned by the run.

---

**enhancements_put_cass_ncoa**

*Replace all attributes of this CASS/NCOA Enhancement*

---

Description

Replace all attributes of this CASS/NCOA Enhancement

Usage

```python
enhancements_put_cass_ncoa(
    id,
    name,
    source,
    schedule = NULL,
    parent_id = NULL,
    notifications = NULL,
    destination = NULL,
    column_mapping = NULL,
    use_default_column_mapping = NULL,
    perform_ncoa = NULL,
    ncoa_credential_id = NULL,
    output_level = NULL,
    limiting_sql = NULL
)
```

Arguments

- **id**: integer required. The ID for the enhancement.
- **name**: string required. The name of the enhancement job.
- **source**: list required. A list containing the following elements:
databaseTable list. A list containing the following elements:

- schema string, The schema name of the source table.
- table string, The name of the source table.
- remoteHostId integer, The ID of the database host for the table.
- credentialId integer, The id of the credentials to be used when performing the enhancement.
- multipartKey array, The source table primary key.

schedule list optional. A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

databaseTable list. A list containing the following elements:

- schema string, The schema name for the output data.
- table string, The table name for the output data.

column_mapping list optional. A list containing the following elements:

- address1 string, The first address line.
- address2 string, The second address line.
- city string, The city of an address.
- state string, The state of an address.
- zip string, The zip code of an address.
name string, The full name of the resident at this address. If needed, separate multiple columns with ‘+', e.g. 'first_name+last_name'

company string, The name of the company located at this address.

use_default_column_mapping
boolean optional. Defaults to true, where the existing column mapping on the input table will be used. If false, a custom column mapping must be provided.

perform_ncoa
boolean optional. Whether to update addresses for records matching the National Change of Address (NCOA) database.

ncoa_credential_id
integer optional. Credential to use when performing NCOA updates. Required if 'performNcoa' is true.

output_level
string optional. The set of fields persisted by a CASS or NCOA enhancement. For CASS enhancements, one of 'cass' or 'all'. For NCOA enhancements, one of 'cass', 'ncoa', 'coalesced' or 'all'. By default, all fields will be returned.

limiting_sql
string optional. The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').

Value

A list containing the following elements:

id integer, The ID for the enhancement.

name string, The name of the enhancement job.

type string, The type of the enhancement (e.g CASS-NCOA)

createdAt string, The time this enhancement was created.

updatedAt string, The time the enhancement was last updated.

author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

state string, The status of the enhancement’s last run

schedule list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

parentId integer, Parent ID that triggers this enhancement.

notifications list, A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
• successEmailSubject string, Custom subject line for success e-mail.
• successEmailBody string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

source list, A list containing the following elements:
• databaseTable list. A list containing the following elements:
  – schema string, The schema name of the source table.
  – table string, The name of the source table.
  – remoteHostId integer, The ID of the database host for the table.
  – credentialId integer, The id of the credentials to be used when performing the enhancement.
  – multipartKey array, The source table primary key.

destination list, A list containing the following elements:
• databaseTable list. A list containing the following elements:
  – schema string, The schema name for the output data.
  – table string, The table name for the output data.

columnMapping list, A list containing the following elements:
• address1 string, The first address line.
• address2 string, The second address line.
• city string, The city of an address.
• state string, The state of an address.
• zip string, The zip code of an address.
• name string, The full name of the resident at this address. If needed, separate multiple columns with ‘+’, e.g. 'first_name+last_name'
• company string, The name of the company located at this address.
useDefaultColumnMapping
    boolean, Defaults to true, where the existing column mapping on the input table will be used. If false, a custom column mapping must be provided.
performNcoa
    boolean, Whether to update addresses for records matching the National Change of Address (NCOA) database.
ncoaCredentialId
    integer, Credential to use when performing NCOA updates. Required if 'performNcoa' is true.
outputLevel
    string, The set of fields persisted by a CASS or NCOA enhancement. For CASS enhancements, one of 'cass' or 'all.' For NCOA enhancements, one of 'cass', 'ncoa', 'coalesced' or 'all'. By default, all fields will be returned.
limitingSQL
    string, The limiting SQL for the source table. "WHERE" should be omitted (e.g. state=’IL’).
archived
    string, The archival status of the requested item(s).

enhancements_put_cass_ncoa_archive

Update the archive status of this object

Description
Update the archive status of this object

Usage
enhancements_put_cass_ncoa_archive(id, status)

Arguments
id
    integer required. The ID of the object.
status
    boolean required. The desired archived status of the object.

Value
A list containing the following elements:

id
    integer, The ID for the enhancement.
name
    string, The name of the enhancement job.
type
    string, The type of the enhancement (e.g CASS-NCOA)
createdAt
    string, The time this enhancement was created.
updatedAt
    string, The time the enhancement was last updated.
author
    list, A list containing the following elements:
    • id integer, The ID of this user.
    • name string, This user's name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

state string, The status of the enhancement’s last run

schedule list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

parentId integer, Parent ID that triggers this enhancement.

notifications list, A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  • successOn boolean, If success email notifications are on.
  • failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

source list, A list containing the following elements:
  • databaseTable list. A list containing the following elements:
    – schema string, The schema name of the source table.
    – table string, The name of the source table.
    – remoteHostId integer, The ID of the database host for the table.
    – credentialHostId integer, The id of the credentials to be used when performing the enhancement.
# enhancements_put_cass_ncoa_projects

**Add a CASS/NCOA Enhancement to a project**

## Description

Add a CASS/NCOA Enhancement to a project

## Usage

```python
enhancements_put_cass_ncoa_projects(id, project_id)
```
enhancements_put_cass_ncoa_shares_groups

Arguments

id  integer required. The ID of the CASS/NCOA Enhancement.
project_id  integer required. The ID of the project.

Value

An empty HTTP response

Description

Set the permissions groups has on this object

Usage

enhancements_put_cass_ncoa_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)

Arguments

id  integer required. The ID of the resource that is shared.
group_ids  array required. An array of one or more group IDs.
permission_level  string required. Options are: "read", "write", or "manage".
share_email_body  string optional. Custom body text for e-mail sent on a share.
send_shared_email  boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers  list, A list containing the following elements:
    • users array,
    • groups array,

writers  list, A list containing the following elements:
Set the permissions users have on this object

**Description**

Set the permissions users have on this object

**Usage**

```python
enhancements_put_cass_ncoa_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```

**Arguments**

- **id**
  - integer required. The ID of the resource that is shared.
- **user_ids**
  - array required. An array of one or more user IDs.
- **permission_level**
  - string required. Options are: "read", "write", or "manage".
- **share_email_body**
  - string optional. Custom body text for e-mail sent on a share.
- **send_shared_email**
  - boolean optional. Send email to the recipients of a share.
Value

A list containing the following elements:

readers list, A list containing the following elements:
• users array,
• groups array,

writers list, A list containing the following elements:
• users array,
• groups array,

owners list, A list containing the following elements:
• users array,
• groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

enhancements_put_civis_data_match

Replace all attributes of this Civis Data Match Enhancement

---

Description

Replace all attributes of this Civis Data Match Enhancement

Usage

```python
eenhancements_put_civis_data_match(
    id,
    name,
    input_field_mapping,
    input_table,
    match_target_id,
    output_table,
    schedule = NULL,
    parent_id = NULL,
    notifications = NULL,
    max_matches = NULL,
    threshold = NULL,
    archived = NULL
)
```
**Arguments**

- **id**
  - integer required. The ID for the enhancement.

- **name**
  - string required. The name of the enhancement job.

- **input_field_mapping**
  - list required. The column mapping for the input table. See `/enhancements/field_mapping` for list of valid fields.

- **input_table**
  - list required. A list containing the following elements:
    - `databaseName` string, The Redshift database name for the table.
    - `schema` string, The schema name for the table.
    - `table` string, The table name.

- **match_target_id**
  - integer required. The ID of the Civis Data match target. See `/match_targets` for IDs.

- **output_table**
  - list required. A list containing the following elements:
    - `databaseName` string, The Redshift database name for the table.
    - `schema` string, The schema name for the table.
    - `table` string, The table name.

- **schedule**
  - list optional. A list containing the following elements:
    - `scheduled` boolean, If the item is scheduled.
    - `scheduledDays` array, Day based on numeric value starting at 0 for Sunday.
    - `scheduledHours` array, Hours of the day it is scheduled on.
    - `scheduledMinutes` array, Minutes of the day it is scheduled on.
    - `scheduledRunsPerHour` integer, Alternative to scheduled minutes, number of times to run per hour.

- **parent_id**
  - integer optional. Parent ID that triggers this enhancement.

- **notifications**
  - list optional. A list containing the following elements:
    - `urls` array, URLs to receive a POST request at job completion
    - `successEmailSubject` string, Custom subject line for success e-mail.
    - `successEmailBody` string, Custom body text for success e-mail, written in Markdown.
    - `successEmailAddresses` array, Addresses to notify by e-mail when the job completes successfully.
    - `successEmailFromName` string, Name from which success emails are sent; defaults to "Civis."
    - `successEmailReplyTo` string, Address for replies to success emails; defaults to the author of the job.
    - `failureEmailAddresses` array, Addresses to notify by e-mail when the job fails.
    - `stallWarningMinutes` integer, Stall warning emails will be sent after this amount of minutes.
    - `successOn` boolean, If success email notifications are on.
    - `failureOn` boolean, If failure email notifications are on.
max_matches integer optional. The maximum number of matches per record in the input table to return. Must be between 0 and 10. 0 returns all matches.

threshold number optional. The score threshold (between 0 and 1). Matches below this threshold will not be returned.

archived boolean optional. Whether the Civis Data Match Job has been archived.

Value

A list containing the following elements:

id integer, The ID for the enhancement.

name string, The name of the enhancement job.

type string, The type of the enhancement (e.g CASS-NCOA)

createdAt string, The time this enhancement was created.

updatedAt string, The time the enhancement was last updated.

author list, A list containing the following elements:

  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

state string, The status of the enhancement’s last run

schedule list, A list containing the following elements:

  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

parentId integer, Parent ID that triggers this enhancement.

notifications list, A list containing the following elements:

  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user's name.
• username string, This user's username.
• initials string, This user's initials.
• online boolean, Whether this user is online.

inputFieldMapping list, The column mapping for the input table. See /enhancements/field_mapping for list of valid fields.

inputTable list, A list containing the following elements:
• databaseName string, The Redshift database name for the table.
• schema string, The schema name for the table.
• table string, The table name.

matchTargetId integer, The ID of the Civis Data match target. See /match_targets for IDs.

outputTable list, A list containing the following elements:
• databaseName string, The Redshift database name for the table.
• schema string, The schema name for the table.
• table string, The table name.

maxMatches integer, The maximum number of matches per record in the input table to return. Must be between 0 and 10. 0 returns all matches.

threshold number, The score threshold (between 0 and 1). Matches below this threshold will not be returned.

archived boolean, Whether the Civis Data Match Job has been archived.

lastRun list, A list containing the following elements:
• id integer,
• state string,
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.
• error string, The error message for this run, if present.
enhancements_put_civis_data_match_archive

Update the archive status of this object

Description

Update the archive status of this object

Usage

enhancements_put_civis_data_match_archive(id, status)

Arguments

id  integer required. The ID of the object.
status  boolean required. The desired archived status of the object.

Value

A list containing the following elements:
id  integer, The ID for the enhancement.
name  string, The name of the enhancement job.
type  string, The type of the enhancement (e.g. CASS-NCOA)
createdAt  string, The time this enhancement was created.
updatedAt  string, The time the enhancement was last updated.
author  list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
state  string, The status of the enhancement’s last run
schedule  list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.
parentId  integer, Parent ID that triggers this enhancement.
notifications  list, A list containing the following elements:
• urls array, URLs to receive a POST request at job completion
• successEmailSubject string, Custom subject line for success e-mail.
• successEmailBody string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user's username.
• initials string, This user's initials.
• online boolean, Whether this user is online.

inputFieldMapping list, The column mapping for the input table. See /enhancements/field_mapping for list of valid fields.

inputTable list, A list containing the following elements:
• databaseName string, The Redshift database name for the table.
• schema string, The schema name for the table.
• table string, The table name.

matchTargetId integer, The ID of the Civis Data match target. See /match_targets for IDs.

outputTable list, A list containing the following elements:
• databaseName string, The Redshift database name for the table.
• schema string, The schema name for the table.
• table string, The table name.

maxMatches integer, The maximum number of matches per record in the input table to return. Must be between 0 and 10. 0 returns all matches.

threshold number, The score threshold (between 0 and 1). Matches below this threshold will not be returned.

archived boolean, Whether the Civis Data Match Job has been archived.

lastRun list, A list containing the following elements:
• id integer,
• state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

---

**enhancements_put_civis_data_match_projects**

*Add a Civis Data Match Enhancement to a project*

**Description**

Add a Civis Data Match Enhancement to a project

**Usage**

`enhancements_put_civis_data_match_projects(id, project_id)`

**Arguments**

- `id` integer required. The ID of the Civis Data Match Enhancement.
- `project_id` integer required. The ID of the project.

**Value**

An empty HTTP response

---

**enhancements_put_civis_data_match_shares_groups**

*Set the permissions groups has on this object*

**Description**

Set the permissions groups has on this object

**Usage**

```python
enhancements_put_civis_data_match_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```
**enhancements_put_civis_data_match_shares_users**

Set the permissions users have on this object

**Arguments**

- **id** integer required. The ID of the resource that is shared.
- **group_ids** array required. An array of one or more group IDs.
- **permission_level** string required. Options are: "read", "write", or "manage".
- **share_email_body** string optional. Custom body text for e-mail sent on a share.
- **send_shared_email** boolean optional. Send email to the recipients of a share.

**Value**

A list containing the following elements:

- **readers** list, A list containing the following elements:
  - **users array**,
  - **groups array**,
- **writers** list, A list containing the following elements:
  - **users array**,
  - **groups array**,
- **owners** list, A list containing the following elements:
  - **users array**,
  - **groups array**,
- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

**Description**

Set the permissions users have on this object

**Usage**

```python
enhancements_put_civis_data_match_shares_users(
    id, 
    user_ids, 
    permission_level, 
    share_email_body = NULL, 
    send_shared_email = NULL 
)
```
Arguments

id integer required. The ID of the resource that is shared.
user_ids array required. An array of one or more user IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers list, A list containing the following elements:
   • users array,
   • groups array,

writers list, A list containing the following elements:
   • users array,
   • groups array,

owners list, A list containing the following elements:
   • users array,
   • groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

Replace all attributes of this Geocode Enhancement

Description

Replace all attributes of this Geocode Enhancement
Usage

```python
enhancements_put_geocode(
    id,
    name,
    remote_host_id,
    credential_id,
    source_schema_and_table,
    schedule = NULL,
    parent_id = NULL,
    notifications = NULL,
    multipart_key = NULL,
    limiting_sql = NULL,
    target_schema = NULL,
    target_table = NULL,
    country = NULL,
    provider = NULL,
    output_address = NULL
)
```

Arguments

- `id` integer required. The ID for the enhancement.
- `name` string required. The name of the enhancement job.
- `remote_host_id` integer required. The ID of the remote host.
- `credential_id` integer required. The ID of the remote host credential.
- `source_schema_and_table` string required. The source database schema and table.
- `schedule` list optional. A list containing the following elements:
  - `scheduled` boolean, If the item is scheduled.
  - `scheduledDays` array, Day based on numeric value starting at 0 for Sunday.
  - `scheduledHours` array, Hours of the day it is scheduled on.
  - `scheduledMinutes` array, Minutes of the day it is scheduled on.
  - `scheduledRunsPerHour` integer, Alternative to scheduled minutes, number of times to run per hour.
- `parent_id` integer optional. Parent ID that triggers this enhancement.
- `notifications` list optional. A list containing the following elements:
  - `urls` array, URLs to receive a POST request at job completion
  - `successEmailSubject` string, Custom subject line for success e-mail.
  - `successEmailBody` string, Custom body text for success e-mail, written in Markdown.
  - `successEmailAddresses` array, Addresses to notify by e-mail when the job completes successfully.
  - `successEmailFromName` string, Name from which success emails are sent; defaults to "Civis."
enhancements_put_geocode

- successEmailReplyTo string. Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array. Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer. Stall warning emails will be sent after this amount of minutes.
- successOn boolean. If success email notifications are on.
- failureOn boolean. If failure email notifications are on.

multipart_key array optional. The source table primary key.
limiting_sql string optional. The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').
target_schema string optional. The output table schema.
target_table string optional. The output table name.
country string optional. The country of the addresses to be geocoded; either 'us' or 'ca'.
provider string optional. The geocoding provider; one of postgis, nominatim, and geocoder_ca.
output_address boolean optional. Whether to output the parsed address. Only guaranteed for the 'postgis' provider.

Value

A list containing the following elements:

- id integer, The ID for the enhancement.
- name string, The name of the enhancement job.
- type string, The type of the enhancement (e.g CASS-NCOA)
- createdAt string, The time this enhancement was created.
- updatedAt string, The time the enhancement was last updated.
- author list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.
- state string, The status of the enhancement’s last run
- schedule list, A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.
- parentId integer, Parent ID that triggers this enhancement.
enhancements_put_geocode

notifications list, A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  • successOn boolean, If success email notifications are on.
  • failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

remoteHostId integer, The ID of the remote host.

credentialId integer, The ID of the remote host credential.

sourceSchemaAndTable string, The source database schema and table.

multipartKey array, The source table primary key.

limitingSQL string, The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').

targetSchema string, The output table schema.

targetTable string, The output table name.

country string, The country of the addresses to be geocoded; either ‘us’ or ‘ca’.

provider string, The geocoding provider; one of postgis, nominatim, and geocoder_ca.

outputAddress boolean, Whether to output the parsed address. Only guaranteed for the 'postgis' provider.

archived string, The archival status of the requested item(s).
**enhancements_put_geocode_archive**

*Update the archive status of this object*

**Description**

Update the archive status of this object

**Usage**

`enhancements_put_geocode_archive(id, status)`

**Arguments**

- **id**  
  integer required. The ID of the object.

- **status**  
  boolean required. The desired archived status of the object.

**Value**

A list containing the following elements:

- **id**  
  integer, The ID for the enhancement.

- **name**  
  string, The name of the enhancement job.

- **type**  
  string, The type of the enhancement (e.g. CASS-NCOA)

- **createdAt**  
  string, The time this enhancement was created.

- **updatedAt**  
  string, The time the enhancement was last updated.

- **author**  
  list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.

- **state**  
  string, The status of the enhancement’s last run

- **schedule**  
  list, A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

- **parentId**  
  integer, Parent ID that triggers this enhancement.

- **notifications**  
  list, A list containing the following elements:
- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
- id integer, The ID of this user.
- name string, This user’s name.
- username string, This user’s username.
- initials string, This user’s initials.
- online boolean, Whether this user is online.

remoteHostId integer, The ID of the remote host.
credentialId integer, The ID of the remote host credential.
sourceSchemaAndTable string, The source database schema and table.
multipartKey array, The source table primary key.
limitingSQL string, The limiting SQL for the source table. "WHERE" should be omitted (e.g. state='IL').
targetSchema string, The output table schema.
targetTable string, The output table name.
country string, The country of the addresses to be geocoded; either 'us' or 'ca'.
provider string, The geocoding provider; one of postgis, nominatim, and geocoder_ca.
outputAddress boolean, Whether to output the parsed address. Only guaranteed for the 'postgis' provider.
archived string, The archival status of the requested item(s).
**enhancements_put_geocode_projects**

*Add a Geocode Enhancement to a project*

**Description**

Add a Geocode Enhancement to a project

**Usage**

```python
enhancements_put_geocode_projects(id, project_id)
```

**Arguments**

- **id**: integer required. The ID of the Geocode Enhancement.
- **project_id**: integer required. The ID of the project.

**Value**

An empty HTTP response

---

**enhancements_put_geocode_shares_groups**

*Set the permissions groups has on this object*

**Description**

Set the permissions groups has on this object

**Usage**

```python
enhancements_put_geocode_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```
**Arguments**

- **id** integer required. The ID of the resource that is shared.
- **group_ids** array required. An array of one or more group IDs.
- **permission_level** string required. Options are: "read", "write", or "manage".
- **share_email_body** string optional. Custom body text for e-mail sent on a share.
- **send_shared_email** boolean optional. Send email to the recipients of a share.

**Value**

A list containing the following elements:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,
- **writers** list, A list containing the following elements:
  - users array,
  - groups array,
- **owners** list, A list containing the following elements:
  - users array,
  - groups array,
- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**enhancements_put_geocode_shares_users**

*Set the permissions users have on this object*

**Description**

Set the permissions users have on this object

**Usage**

```python
enhancements_put_geocode_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```
exports_get_files_csv

Arguments

id integer required. The ID of the resource that is shared.
user_ids array required. An array of one or more user IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers list, A list containing the following elements:
  • users array,
  • groups array,
writers list, A list containing the following elements:
  • users array,
  • groups array,
owners list, A list containing the following elements:
  • users array,
  • groups array,
totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

Description

Get a CSV Export

Usage

exports_get_files_csv(id)

Arguments

id integer required.
### Value

A list containing the following elements:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer, The ID of this Csv Export job.</td>
</tr>
<tr>
<td>name</td>
<td>string, The name of this Csv Export job.</td>
</tr>
<tr>
<td>source</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>- sql string, The SQL query for this Csv Export job</td>
</tr>
<tr>
<td></td>
<td>- remoteHostId integer, The ID of the destination database host.</td>
</tr>
<tr>
<td></td>
<td>- credentialId integer, The ID of the credentials for the destination database.</td>
</tr>
<tr>
<td>destination</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>- filenamePrefix string, The prefix of the name of the file returned to the user.</td>
</tr>
<tr>
<td></td>
<td>- storagePath list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>includeHeader</td>
<td>boolean, A boolean value indicating whether or not the header should be included. Defaults to true.</td>
</tr>
<tr>
<td>compression</td>
<td>string, The compression of the output file. Valid arguments are &quot;gzip&quot; and &quot;none&quot;. Defaults to &quot;gzip&quot;.</td>
</tr>
<tr>
<td>columnDelimiter</td>
<td>string, The column delimiter for the output file. Valid arguments are &quot;comma&quot;, &quot;tab&quot;, and &quot;pipe&quot;. Defaults to &quot;comma&quot;.</td>
</tr>
<tr>
<td>hidden</td>
<td>boolean, A boolean value indicating whether or not this request should be hidden. Defaults to false.</td>
</tr>
<tr>
<td>forceMultifile</td>
<td>boolean, Whether or not the csv should be split into multiple files. Default: false</td>
</tr>
<tr>
<td>maxFileSize</td>
<td>integer, The max file size, in MB, created files will be. Only available when force_multifile is true.</td>
</tr>
</tbody>
</table>
exports_list

List

Description

List

Usage

exports_list(
    type = NULL,
    author = NULL,
    status = NULL,
    hidden = NULL,
    archived = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)

Arguments

- **type**: string optional. If specified, return exports of these types. It accepts a comma-separated list, possible values are 'database' and 'gdoc'.
- **author**: string optional. If specified, return exports from this author. It accepts a comma-separated list of author ids.
- **status**: string optional. If specified, returns export with one of these statuses. It accepts a comma-separated list, possible values are 'running', 'failed', 'succeeded', 'idle', 'scheduled'.
- **hidden**: boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
- **archived**: string optional. The archival status of the requested item(s).
- **limit**: integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
- **page_num**: integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order**: string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at, last_run.updated_at.
- **order_dir**: string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.
Value

An array containing the following fields:

- **id**: integer, The ID for this export.
- **name**: string, The name of this export.
- **type**: string, The type of export.
- **createdAt**: string, The creation time for this export.
- **updatedAt**: string, The last modification time for this export.
- **state**: string,
- **lastRun**: list, A list containing the following elements:
  - **id**: integer,
  - **state**: string,
  - **createdAt**: string, The time that the run was queued.
  - **startedAt**: string, The time that the run started.
  - **finishedAt**: string, The time that the run completed.
  - **error**: string, The error message for this run, if present.
- **author**: list, A list containing the following elements:
  - **id**: integer, The ID of this user.
  - **name**: string, This user’s name.
  - **username**: string, This user’s username.
  - **initials**: string, This user’s initials.
  - **online**: boolean, Whether this user is online.

exports_patch_files_csv

*Update some attributes of this CSV Export*

Description

Update some attributes of this CSV Export

Usage

```python
exports_patch_files_csv(
    id,
    name = NULL,
    source = NULL,
    destination = NULL,
    include_header = NULL,
    compression = NULL,
    column_delimiter = NULL,
    hidden = NULL,
    force_multifile = NULL,
    max_file_size = NULL
)
```
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of this Csv Export job.</td>
</tr>
<tr>
<td>name</td>
<td>string optional. The name of this Csv Export job.</td>
</tr>
<tr>
<td>source</td>
<td>list optional. A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• sql string, The SQL query for this Csv Export job</td>
</tr>
<tr>
<td></td>
<td>• remoteHostId integer, The ID of the destination database host.</td>
</tr>
<tr>
<td></td>
<td>• credentialId integer, The ID of the credentials for the destination database.</td>
</tr>
<tr>
<td>destination</td>
<td>list optional. A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• filenamePrefix string, The prefix of the name of the file returned to the user.</td>
</tr>
<tr>
<td></td>
<td>• storagePath list. A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>– filePath string, The path within the bucket where the exported file will be saved. E.g. the file_path for &quot;s3://mybucket/files/all/&quot; would be &quot;/files/all/&quot;</td>
</tr>
<tr>
<td></td>
<td>– remoteHostId integer, The ID of the destination database host.</td>
</tr>
<tr>
<td></td>
<td>– credentialId integer, The ID of the credentials for the destination storage host.</td>
</tr>
<tr>
<td></td>
<td>– existingFiles string, Notifies the job of what to do in the case that the exported file already exists at the provided path. One of: fail, append, overwrite. Default: fail. If &quot;append&quot; is specified, the new file will always be added to the provided path. If &quot;overwrite&quot; is specified, all existing files at the provided path will be deleted and the new file will be added. By default, or if &quot;fail&quot; is specified, the export will fail if a file exists at the provided path.</td>
</tr>
<tr>
<td>include_header</td>
<td>boolean optional. A boolean value indicating whether or not the header should be included. Defaults to true.</td>
</tr>
<tr>
<td>compression</td>
<td>string optional. The compression of the output file. Valid arguments are &quot;gzip&quot; and &quot;none&quot;. Defaults to &quot;gzip&quot;.</td>
</tr>
<tr>
<td>column_delimiter</td>
<td>string optional. The column delimiter for the output file. Valid arguments are &quot;comma&quot;, &quot;tab&quot;, and &quot;pipe&quot;. Defaults to &quot;comma&quot;.</td>
</tr>
<tr>
<td>hidden</td>
<td>boolean optional. A boolean value indicating whether or not this request should be hidden. Defaults to false.</td>
</tr>
<tr>
<td>force_multifile</td>
<td>boolean optional. Whether or not the csv should be split into multiple files. Default: false</td>
</tr>
<tr>
<td>max_file_size</td>
<td>integer optional. The max file size, in MB, created files will be. Only available when force_multifile is true.</td>
</tr>
</tbody>
</table>

Value

A list containing the following elements:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer, The ID of this Csv Export job.</td>
</tr>
<tr>
<td>name</td>
<td>string, The name of this Csv Export job.</td>
</tr>
</tbody>
</table>
exports_post_files_csv

Create a CSV Export

Description

Create a CSV Export

source list, A list containing the following elements:
• sql string, The SQL query for this Csv Export job
• remoteHostId integer, The ID of the destination database host.
• credentialId integer, The ID of the credentials for the destination database.

destination list, A list containing the following elements:
• filenamePrefix string, The prefix of the name of the file returned to the user.
• storagePath list. A list containing the following elements:
  – filePath string, The path within the bucket where the exported file will be saved. E.g. the file path for "s3://mybucket/files/all/" would be "/files/all/"
  – storageHostId integer, The ID of the destination storage host.
  – credentialId integer, The ID of the credentials for the destination storage host.
  – existingFiles string, Notifies the job of what to do in the case that the exported file already exists at the provided path. One of: fail, append, overwrite. Default: fail. If "append" is specified, the new file will always be added to the provided path. If "overwrite" is specified, all existing files at the provided path will be deleted and the new file will be added. By default, or if "fail" is specified, the export will fail if a file exists at the provided path.

includeHeader boolean, A boolean value indicating whether or not the header should be included. Defaults to true.

compression string, The compression of the output file. Valid arguments are "gzip" and "none". Defaults to "gzip".

columnDelimiter string, The column delimiter for the output file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

hidden boolean, A boolean value indicating whether or not this request should be hidden. Defaults to false.

forceMultifile boolean, Whether or not the csv should be split into multiple files. Default: false

maxFileSize integer, The max file size, in MB, created files will be. Only available when force_multifile is true.
Usage

```r
events_post_files_csv(
  source,
  destination,
  name = NULL,
  include_header = NULL,
  compression = NULL,
  column_delimiter = NULL,
  hidden = NULL,
  force_multifile = NULL,
  max_file_size = NULL
)
```

Arguments

- **source** list required. A list containing the following elements:
  - sql string, The SQL query for this Csv Export job
  - remoteHostId integer, The ID of the destination database host.
  - credentialId integer, The ID of the credentials for the destination database.

- **destination** list required. A list containing the following elements:
  - filenamePrefix string, The prefix of the name of the file returned to the user.
  - storagePath list. A list containing the following elements:
    - filePath string, The path within the bucket where the exported file will be saved. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"
    - storageHostId integer, The ID of the destination storage host.
    - credentialId integer, The ID of the credentials for the destination storage host.
    - existingFiles string, Notifies the job of what to do in the case that the exported file already exists at the provided path. One of: fail, append, overwrite. Default: fail. If "append" is specified, the new file will always be added to the provided path. If "overwrite" is specified, all existing files at the provided path will be deleted and the new file will be added. By default, or if "fail" is specified, the export will fail if a file exists at the provided path.

- **name** string optional. The name of this Csv Export job.

- **include_header** boolean optional. A boolean value indicating whether or not the header should be included. Defaults to true.

- **compression** string optional. The compression of the output file. Valid arguments are "gzip" and "none". Defaults to "gzip".

- **column_delimiter** string optional. The column delimiter for the output file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

- **hidden** boolean optional. A boolean value indicating whether or not this request should be hidden. Defaults to false.
exports_post_files_csv

force_multifile
boolean optional. Whether or not the csv should be split into multiple files.
Default: false

max_file_size
integer optional. The max file size, in MB, created files will be. Only available when force_multifile is true.

Value
A list containing the following elements:

- id
  integer, The ID of this Csv Export job.

- name
  string, The name of this Csv Export job.

- source
  list, A list containing the following elements:
    - sql string, The SQL query for this Csv Export job
    - remoteHostId integer, The ID of the destination database host.
    - credentialId integer, The ID of the credentials for the destination database.

- destination
  list, A list containing the following elements:
    - filenamePrefix string, The prefix of the name of the file returned to the user.
    - storagePath list, A list containing the following elements:
      - filePath string, The path within the bucket where the exported file will be saved. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"
      - storageHostId integer, The ID of the destination storage host.
      - credentialId integer, The ID of the credentials for the destination storage host.
      - existingFiles string, Notifies the job of what to do in the case that the exported file already exists at the provided path. One of: fail, append, overwrite. Default: fail. If "append" is specified, the new file will always be added to the provided path. If "overwrite" is specified all existing files at the provided path will be deleted and the new file will be added. By default, or if "fail" is specified, the export will fail if a file exists at the provided path.

- includeHeader
  boolean, A boolean value indicating whether or not the header should be included. Default to true.

- compression
  string, The compression of the output file. Valid arguments are "gzip" and "none". Default to "gzip".

- columnDelimiter
  string, The column delimiter for the output file. Valid arguments are "comma", "tab", and "pipe". Default to "comma".

- hidden
  boolean, A boolean value indicating whether or not this request should be hidden. Default to false.

- forceMultifile
  boolean, Whether or not the csv should be split into multiple files. Default: false

- maxFileSize
  integer, The max file size, in MB, created files will be. Only available when force_multifile is true.
exports_put_files_csv  Replace all attributes of this CSV Export

Description

Replace all attributes of this CSV Export

Usage

exports_put_files_csv(
    id,
    source,
    destination,
    name = NULL,
    include_header = NULL,
    compression = NULL,
    column_delimiter = NULL,
    hidden = NULL,
    force_multifile = NULL,
    max_file_size = NULL
)

Arguments

id          integer required. The ID of this Csv Export job.
source      list required. A list containing the following elements:
            • sql string, The SQL query for this Csv Export job
            • remoteHostId integer, The ID of the destination database host.
            • credentialId integer, The ID of the credentials for the destination database.
destination list required. A list containing the following elements:
            • filenamePrefix string, The prefix of the name of the file returned to the user.
            • storagePath list. A list containing the following elements:
                • filePath string, The path within the bucket where the exported file will be saved. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"
                • storageHostId integer, The ID of the destination storage host.
                • credentialId integer, The ID of the credentials for the destination storage host.
                • existingFiles string, Notifies the job of what to do in the case that the exported file already exists at the provided path. One of: fail, append, overwrite. Default: fail. If "append" is specified, the new file will always be added to the provided path. If "overwrite" is specified all existing files at the provided path will be deleted and the new file will be added. By default, or if "fail" is specified, the export will fail if a file exists at the provided path.
name string optional. The name of this Csv Export job.
include_header boolean optional. A boolean value indicating whether or not the header should be included. Defaults to true.
compression string optional. The compression of the output file. Valid arguments are "gzip" and "none". Defaults to "gzip".
column_delimiter string optional. The column delimiter for the output file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".
hidden boolean optional. A boolean value indicating whether or not this request should be hidden. Defaults to false.
force_multifile boolean optional. Whether or not the csv should be split into multiple files. Default: false
max_file_size integer optional. The max file size, in MB, created files will be. Only available when force_multifile is true.

Value
A list containing the following elements:

id integer, The ID of this Csv Export job.
name string, The name of this Csv Export job.
source list, A list containing the following elements:
• sql string, The SQL query for this Csv Export job
• remoteHostId integer, The ID of the destination database host.
• credentialId integer, The ID of the credentials for the destination database.
destination list, A list containing the following elements:
• filenamePrefix string, The prefix of the name of the file returned to the user.
• storagePath list. A list containing the following elements:
  – filePath string, The path within the bucket where the exported file will be saved. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"
  – storageHostId integer, The ID of the destination storage host.
  – credentialId integer, The ID of the credentials for the destination storage host.
  – existingFiles string, Notifies the job of what to do in the case that the exported file already exists at the provided path. One of: fail, append, overwrite. Default: fail. If "append" is specified, the new file will always be added to the provided path. If "overwrite" is specified all existing files at the provided path will be deleted and the new file will be added. By default, or if "fail" is specified, the export will fail if a file exists at the provided path.
includeHeader boolean, A boolean value indicating whether or not the header should be included. Defaults to true.
exports_put_files_csv_archive

Update the archive status of this object

Description

Update the archive status of this object

Usage

exports_put_files_csv_archive(id, status)

Arguments

id
integer required. The ID of the object.

status
boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id
integer, The ID of this Csv Export job.

name
string, The name of this Csv Export job.

source
list, A list containing the following elements:
- sql string, The SQL query for this Csv Export job
- remoteHostId integer, The ID of the destination database host.
- credentialId integer, The ID of the credentials for the destination database.

destination
list, A list containing the following elements:
- filenamePrefix string, The prefix of the name of the file returned to the user.
- storagePath list, A list containing the following elements:
  - filePath string, The path within the bucket where the exported file will be saved. E.g. the `filePath` for "s3://mybucket/files/all/" would be "/files/all/"

compression
string, The compression of the output file. Valid arguments are "gzip" and "none". Defaults to "gzip".

columnDelimiter
string, The column delimiter for the output file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

hidden
boolean, A boolean value indicating whether or not this request should be hidden. Defaults to false.

forceMultifile
boolean, Whether or not the csv should be split into multiple files. Default: false

maxFileSize
integer, The max file size, in MB, created files will be. Only available when force_multifile is true.
- `storageHostId` integer, The ID of the destination storage host.
- `credentialId` integer, The ID of the credentials for the destination storage host.
- `existingFiles` string, Notifies the job of what to do in the case that the exported file already exists at the provided path. One of: fail, append, overwrite. Default: fail. If "append" is specified, the new file will always be added to the provided path. If "overwrite" is specified, all existing files at the provided path will be deleted and the new file will be added. By default, or if "fail" is specified, the export will fail if a file exists at the provided path.

<table>
<thead>
<tr>
<th>includeHeader</th>
<th>boolean, A boolean value indicating whether or not the header should be included. Defaults to true.</th>
</tr>
</thead>
<tbody>
<tr>
<td>compression</td>
<td>string, The compression of the output file. Valid arguments are &quot;gzip&quot; and &quot;none&quot;. Defaults to &quot;gzip&quot;.</td>
</tr>
<tr>
<td>columnDelimiter</td>
<td>string, The column delimiter for the output file. Valid arguments are &quot;comma&quot;, &quot;tab&quot;, and &quot;pipe&quot;. Defaults to &quot;comma&quot;.</td>
</tr>
<tr>
<td>hidden</td>
<td>boolean, A boolean value indicating whether or not this request should be hidden. Defaults to false.</td>
</tr>
<tr>
<td>forceMultifile</td>
<td>boolean, Whether or not the csv should be split into multiple files. Default: false</td>
</tr>
<tr>
<td>maxFileSize</td>
<td>integer, The max file size, in MB, created files will be. Only available when <code>force_multifile</code> is true.</td>
</tr>
</tbody>
</table>

---

**feature_flags_delete_groups**

*Deactivate a feature for a group*

**Description**

Deactivate a feature for a group

**Usage**

`feature_flags_delete_groups(flag_name, group_id)`

**Arguments**

<table>
<thead>
<tr>
<th>flag_name</th>
<th>string required. The feature flag name.</th>
</tr>
</thead>
<tbody>
<tr>
<td>group_id</td>
<td>integer required. Group ID.</td>
</tr>
</tbody>
</table>

**Value**

An empty HTTP response
feature_flags_delete_users

*Deactivate a feature for a user*

**Description**

Deactivate a feature for a user

**Usage**

`feature_flags_delete_users(flag_name, user_id)`

**Arguments**

- `flag_name`  
  string required. The feature flag name.
- `user_id`  
  integer required. The user ID.

**Value**

An empty HTTP response

feature_flags_get

*Show a feature flag*

**Description**

Show a feature flag

**Usage**

`feature_flags_get(name)`

**Arguments**

- `name`  
  string required. The name of the feature flag.

**Value**

A list containing the following elements:

- `name`  
  string, The name of the feature flag
- `organizations`  
  array, An array containing the following fields:
  - `id` integer, Organization ID
  - `name` string, Organization name
  - `slug` string, Organization slug
groups array, An array containing the following fields:
  • id integer, Group ID
  • name string, Group name
  • slug string, Group slug
  • users array, Users within the group

users array, An array containing the following fields:
  • id integer, User ID
  • username string, Username

percentage integer, The target percentage of users who should have this feature flag enabled.

---

**Description**

List feature flags

**Usage**

feature_flags_list()  

**Value**

An array containing the following fields:

- **name** string, The name of the feature flag.
- **userCount** integer, The number of users with this feature flag enabled.
- **description** string,
- **team** string,
- **jira** string,
- **added** string,
- **groupCount** integer,
- **organizationCount** integer,
- **percentage** integer, The target percentage of users who should have this feature flag enabled.
- **activeForMe** boolean, Whether the feature flag is active for the current user.
feature_flags_put_groups

Activate a feature for a group

Description

Activate a feature for a group

Usage

feature_flags_put_groups(flag_name, group_id)

Arguments

flag_name     string required. The feature flag name.
group_id      integer required. Group ID.

Value

A list containing the following elements:

name     string, The name of the feature.
userCount integer, The number of users with this feature flag enabled.
description string,
team     string,
jira     string,
added    string,
groupCount integer,
organizationCount integer,
percentage integer, The target percentage of users who should have this feature flag enabled.
activeForMe boolean, Whether the feature flag is active for the current user.
feature_flags_put_users

Activate a feature for a user

Description

Activate a feature for a user

Usage

feature_flags_put_users(flag_name, user_id)

Arguments

flag_name string required. The feature flag name.
user_id integer required. The user ID.

Value

A list containing the following elements:

name string, The name of the feature flag.
userCount integer, The number of users with this feature flag enabled.
description string,
team string,
jira string,
added string,
groupCount integer,
organizationCount integer,
percentage integer, The target percentage of users who should have this feature flag enabled.
activeForMe boolean, Whether the feature flag is active for the current user.
**fetch_all**

*Retrieve all results from a paginated endpoint*

**Description**

Retrieve all results from a paginated endpoint

**Usage**

```r
fetch_all(fn, ...)
```

**Arguments**

- `fn` The API function to be called.
- `...` Arguments passed to `fn`.

**Value**

A list with the concatenated results of each page of `fn`.

**See Also**

Other pagination: `fetch_until()`

**Examples**

```r
## Not run:
columns <- fetch_all(tables_list_columns, id = 123)
column_names <- columns %>% purrr::map_chr("name")
## End(Not run)
```

**fetch_and_generate_client**

*Fetches and generates the client in generated_client.R*

**Description**

Fetches and generates the client in generated_client.R

**Usage**

```r
fetch_and_generate_client()
```

**Details**

Skips autogeneration on windows with R < 3.4.0 and if `R_CLIENT_DEV == "TRUE"`. A valid `CIVIS_API_KEY` must be set.
fetch_logs

Fetch job logs from the Civis Platform

Description

Fetch job logs from the Civis Platform

Usage

fetch_logs(object, limit = 100, ...)

Arguments

object: A civis_api, civis_error, or civis_ml object.
limit: The number of log lines to fetch, the default is 100.
... Extra parameters passed to methods, currently unused.

Value

A list of log messages with class civis_logs.

Examples

```r
## Not run:
m <- civis_ml_fetch_existing(123)
fetch_logs(m)

import <- write_civis(iris, "scratch.mytest")
fetch_logs(import)

query_run <- query_civis(123)
fetch_logs(query_run)

e <- tryCatch(read_civis("asdf"), error = function(e) e)
fetch_logs(e)

## End(Not run)
```
## fetch_oos_scores

### Retrieve out of sample/fold predictions

**Description**

Retrieve out of sample/fold predictions

**Usage**

`fetch_oos_scores(model, ...)`

**Arguments**

- `model` A `civis_ml` model.
- `...` Parameters passed to `read.csv`.

**Details**

Returns the out of sample (or out of fold) predictions for each observation. For a multiclass model, a data frame is returned with one column of predictions for each class.

**Value**

A `data.frame` with out of sample/fold predictions for each row of the training data, and containing an additional column with a primary key.

**See Also**

civis_ml

## fetch_output_file_ids

### Return output from a civis_script

**Description**

Return output from a `civis_script`.

**Usage**

`fetch_output_file_ids(x, regex = NULL)`

`fetch_output(x, regex = NULL)`

**Arguments**

- `x` `civis_script`
- `regex` string, regex used to match the run output name.
fetch_predictions

Retrieve predictions from a CivisML prediction job

Description

Retrieve predictions from a CivisML prediction job

Usage

fetch_predictions(x, ...)

Arguments

x  civis_ml_prediction object from predict.civis_ml
...

Arguments passed to read.csv

Details

Predictions can also be downloaded as a csv directly using download_civis (see examples).
Value

A data.frame of predictions containing an additional column with a primary key. For a multiclass model, a data frame is returned with one column of predictions for each class.

Examples

```r
## Not run:
m <- civis_ml("path/to/file.csv", model_type = "sparse_logistic", 
    dependent_variable = "Species")
pred_job <- predict(m, newdata = "path/to/newdata.csv")
yhat <- fetch_predictions(pred_job)

# download instead:
download_civis(pred_job$model_info$output_file_ids, path = "my_predictions.csv")

## End(Not run)
```

fetch_until

Retrieve some results from a paginated endpoint

Description

fetch_until will retrieve paginated results until a condition is met. This is useful when searching for a particular value or record.

Usage

fetch_until(fn, .until, ...)

Arguments

- `fn`: The API function to be called.
- `.until`: A function which returns a boolean value. `.until` will be called with each item from the API response. When `.until` returns TRUE iteration will stop and fetch_until will return all responses accumulated so far.
- `...`: Arguments passed to `fn`.

Value

A list with the concatenated results of each page of `fn`.

See Also

Other pagination: `fetch_all()`
## Not run:
columns <- fetch_until(tables_list_columns,
  until = function(x) x == "voterbase_id")

## End(Not run)

---

### Description

Archive a Preprocess CSV (deprecated, use archiving endpoints instead)

### Usage

```r
files_delete_preprocess_csv(id)
```

### Arguments

- **id**: integer required.

### Value

An empty HTTP response

---

### Description

Remove a File from a project

### Usage

```r
files_delete_projects(id, project_id)
```

### Arguments

- **id**: integer required. The ID of the File.
- **project_id**: integer required. The ID of the project.

### Value

An empty HTTP response
files_delete_shares_groups

Revoke the permissions a group has on this object

Description
Revoke the permissions a group has on this object

Usage
files_delete_shares_groups(id, group_id)

Arguments
- id: integer required. The ID of the resource that is shared.
- group_id: integer required. The ID of the group.

Value
An empty HTTP response

files_delete_shares_users

Revoke the permissions a user has on this object

Description
Revoke the permissions a user has on this object

Usage
files_delete_shares_users(id, user_id)

Arguments
- id: integer required. The ID of the resource that is shared.
- user_id: integer required. The ID of the user.

Value
An empty HTTP response
files_get Get details about a file

Description
Get details about a file

Usage
files_get(id, link_expires_at = NULL, inline = NULL)

Arguments
id integer required. The ID of the file.
link_expires_at string optional. The date and time the download link will expire. Must be a time between now and 36 hours from now. Defaults to 30 minutes from now.
inline boolean optional. If true, will return a url that can be displayed inline in HTML

Value
A list containing the following elements:

id integer, The ID of the file.
name string, The file name.
createdAt string, The date and time the file was created.
fileSize integer, The file size.
expiresAt string, The date and time the file will expire. If not specified, the file will expire in 30 days. To keep a file indefinitely, specify null.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
downloadUrl string, A JSON string containing information about the URL of the file.
fileUrl string, The URL that may be used to download the file.
detectedInfo list, A list containing the following elements:
  • includeHeader boolean, A boolean value indicating whether or not the first row of the file is a header row.
  • columnDelimiter string, The column delimiter for the file. One of "comma", "tab", or "pipe".
files_get_preprocess_csv

Get a Preprocess CSV

Description
Get a Preprocess CSV

Usage
files_get_preprocess_csv(id)

Arguments
id integer required.

Value
A list containing the following elements:

id integer, The ID of the job created.

fileId integer, The ID of the file.

inPlace boolean, If true, the file is cleaned in place. If false, a new file ID is created. Defaults to true.

detectTableColumns boolean, If true, detect the table columns in the file including the sql types. If false, skip table column detection. Defaults to false.

forceCharacterSetConversion boolean, If true, the file will always be converted to UTF-8 and any character that cannot be converted will be discarded. If false, the character set conversion will only run if the detected character set is not compatible with UTF-8 (e.g., UTF-8, ASCII).

includeHeader boolean, A boolean value indicating whether or not the first row of the file is a header row. If not provided, will attempt to auto-detect whether a header row is present.

columnDelimiter string, The column delimiter for the file. One of "comma", "tab", or "pipe". If not provided, the column delimiter will be auto-detected.

hidden boolean, The hidden status of the item.
files_list_projects

List the projects a File belongs to

Description

List the projects a File belongs to

Usage

files_list_projects(id, hidden = NULL)

Arguments

id integer required. The ID of the File.
hidden boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id integer, The ID for this project.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
name string, The name of this project.
description string, A description of the project.
users array, An array containing the following fields:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
autoShare boolean,
createdAt string,
updatedAt string,
archived string, The archival status of the requested item(s).
files_list_shares  
List users and groups permissioned on this object

Description
List users and groups permissioned on this object

Usage
files_list_shares(id)

Arguments
id  integer required. The ID of the resource that is shared.

Value
An array containing the following fields:

readers  list, A list containing the following elements:
  • users array,
  • groups array,

writers  list, A list containing the following elements:
  • users array,
  • groups array,

owners  list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares  integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares  integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.
files_patch

Update details about a file

Description
Update details about a file

Usage
files_patch(id, name = NULL, expires_at = NULL)

Arguments
- id integer required. The ID of the file.
- name string optional. The file name. The extension must match the previous exten-
  sion.
- expires_at string optional. The date and time the file will expire.

Value
A list containing the following elements:
- id integer, The ID of the file.
- name string, The file name.
- createdAt string, The date and time the file was created.
- fileSize integer, The file size.
- expiresAt string, The date and time the file will expire. If not specified, the file will expire in 30 days. To keep a file indefinitely, specify null.
- author list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.
- downloadUrl string, A JSON string containing information about the URL of the file.
- fileUrl string, The URL that may be used to download the file.
- detectedInfo list, A list containing the following elements:
  - includeHeader boolean, A boolean value indicating whether or not the first row of the file is a header row.
  - columnDelimiter string, The column delimiter for the file. One of "comma", "tab", or "pipe".
  - compression string, The type of compression of the file. One of "gzip", or "none".
  - tableColumns array, An array of hashes corresponding to the columns in the file. Each hash should have keys for column "name" and "sql_type"
files_patch_preprocess_csv

*Description*

Update some attributes of this Preprocess CSV

*Usage*

```r
files_patch_preprocess_csv(
    id,
    file_id = NULL,
    in_place = NULL,
    detect_table_columns = NULL,
    force_character_set_conversion = NULL,
    include_header = NULL,
    column_delimiter = NULL
)
```

*Arguments*

- **id** integer required. The ID of the job created.
- **file_id** integer optional. The ID of the file.
- **in_place** boolean optional. If true, the file is cleaned in place. If false, a new file ID is created. Defaults to true.
- **detect_table_columns** boolean optional. If true, detect the table columns in the file including the sql types. If false, skip table column detection. Defaults to false.
- **force_character_set_conversion** boolean optional. If true, the file will always be converted to UTF-8 and any character that cannot be converted will be discarded. If false, the character set conversion will only run if the detected character set is not compatible with UTF-8 (e.g., UTF-8, ASCII).
- **include_header** boolean optional. A boolean value indicating whether or not the first row of the file is a header row. If not provided, will attempt to auto-detect whether a header row is present.
- **column_delimiter** string optional. The column delimiter for the file. One of "comma", "tab", or "pipe". If not provided, the column delimiter will be auto-detected.

*Value*

A list containing the following elements:

- **id** integer, The ID of the job created.
files_post

**Description**

Initiate an upload of a file into the platform

**Usage**

files_post(name, expires_at = NULL)

**Arguments**

- **name**: string required. The file name.
- **expires_at**: string optional. The date and time the file will expire. If not specified, the file will expire in 30 days. To keep a file indefinitely, specify null.

**Value**

A list containing the following elements:

- **id**: integer, The ID of the file.
- **name**: string, The file name.
- **createdAt**: string, The date and time the file was created.
- **fileSize**: integer, The file size.
files_post_multipart

Description

Initiate a multipart upload

Usage

files_post_multipart(name, num_parts, expires_at = NULL)

Arguments

name  string required. The file name.
num_parts  integer required. The number of parts in which the file will be uploaded. This parameter determines the number of presigned URLs that are returned.
expires_at  string optional. The date and time the file will expire. If not specified, the file will expire in 30 days. To keep a file indefinitely, specify null.

Value

A list containing the following elements:

id  integer, The ID of the file.
name  string, The file name.
createdAt  string, The date and time the file was created.
fileSize  integer, The file size.
expiresAt  string, The date and time the file will expire. If not specified, the file will expire in 30 days. To keep a file indefinitely, specify null.
uploadUrls  array, An array of URLs that may be used to upload file parts. Use separate PUT requests to complete the part uploads. Links expire after 12 hours.
files_post_multipart_complete

Complete a multipart upload

Description

Complete a multipart upload

Usage

files_post_multipart_complete(id)

Arguments

id integer required. The ID of the file.

Value

An empty HTTP response

files_post_preprocess_csv

Create a Preprocess CSV

Description

Create a Preprocess CSV

Usage

files_post_preprocess_csv(
    file_id,
    in_place = NULL,
    detect_table_columns = NULL,
    force_character_set_conversion = NULL,
    include_header = NULL,
    column_delimiter = NULL,
    hidden = NULL
)
**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>file_id</td>
<td>integer required. The ID of the file.</td>
</tr>
<tr>
<td>in_place</td>
<td>boolean optional. If true, the file is cleaned in place. If false, a new file ID is created. Defaults to true.</td>
</tr>
<tr>
<td>detect_table_columns</td>
<td>boolean optional. If true, detect the table columns in the file including the sql types. If false, skip table column detection. Defaults to false.</td>
</tr>
<tr>
<td>force_character_set_conversion</td>
<td>boolean optional. If true, the file will always be converted to UTF-8 and any character that cannot be converted will be discarded. If false, the character set conversion will only run if the detected character set is not compatible with UTF-8 (e.g., UTF-8, ASCII).</td>
</tr>
<tr>
<td>include_header</td>
<td>boolean optional. A boolean value indicating whether or not the first row of the file is a header row. If not provided, will attempt to auto-detect whether a header row is present.</td>
</tr>
<tr>
<td>column_delimiter</td>
<td>string optional. The column delimiter for the file. One of &quot;comma&quot;, &quot;tab&quot;, or &quot;pipe&quot;. If not provided, the column delimiter will be auto-detected.</td>
</tr>
<tr>
<td>hidden</td>
<td>boolean optional. The hidden status of the item.</td>
</tr>
</tbody>
</table>

**Value**

A list containing the following elements:

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer, The ID of the job created.</td>
</tr>
<tr>
<td>fileId</td>
<td>integer, The ID of the file.</td>
</tr>
<tr>
<td>inPlace</td>
<td>boolean, If true, the file is cleaned in place. If false, a new file ID is created. Defaults to true.</td>
</tr>
<tr>
<td>detectTableColumns</td>
<td>boolean, If true, detect the table columns in the file including the sql types. If false, skip table column detection. Defaults to false.</td>
</tr>
<tr>
<td>forceCharacterSetConversion</td>
<td>boolean, If true, the file will always be converted to UTF-8 and any character that cannot be converted will be discarded. If false, the character set conversion will only run if the detected character set is not compatible with UTF-8 (e.g., UTF-8, ASCII).</td>
</tr>
<tr>
<td>includeHeader</td>
<td>boolean, A boolean value indicating whether or not the first row of the file is a header row. If not provided, will attempt to auto-detect whether a header row is present.</td>
</tr>
<tr>
<td>columnDelimiter</td>
<td>string, The column delimiter for the file. One of &quot;comma&quot;, &quot;tab&quot;, or &quot;pipe&quot;. If not provided, the column delimiter will be auto-detected.</td>
</tr>
<tr>
<td>hidden</td>
<td>boolean, The hidden status of the item.</td>
</tr>
</tbody>
</table>
files_put

Update details about a file

Description
Update details about a file

Usage
files_put(id, name, expires_at)

Arguments
id integer required. The ID of the file.
name string required. The file name. The extension must match the previous extension.
expires_at string required. The date and time the file will expire.

Value
A list containing the following elements:
id integer, The ID of the file.
name string, The file name.
createdAt string, The date and time the file was created.
fileSize integer, The file size.
expiresAt string, The date and time the file will expire. If not specified, the file will expire in 30 days. To keep a file indefinitely, specify null.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user's name.
  • username string, This user's username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
downloadUrl string, A JSON string containing information about the URL of the file.
fileUrl string, The URL that may be used to download the file.
detectedInfo list, A list containing the following elements:
  • includeHeader boolean, A boolean value indicating whether or not the first row of the file is a header row.
  • columnDelimiter string, The column delimiter for the file. One of "comma", "tab", or "pipe".
  • compression string, The type of compression of the file. One of "gzip", or "none".
  • tableColumns array, An array of hashes corresponding to the columns in the file. Each hash should have keys for column "name" and "sql_type"
files_put_preprocess_csv

Replace all attributes of this Preprocess CSV

Description

Replace all attributes of this Preprocess CSV

Usage

files_put_preprocess_csv(
id,
file_id,
in_place = NULL,
detect_table_columns = NULL,
force_character_set_conversion = NULL,
include_header = NULL,
column_delimiter = NULL
)

Arguments

id  integer required. The ID of the job created.
file_id  integer required. The ID of the file.
in_place  boolean optional. If true, the file is cleaned in place. If false, a new file ID is created. Defaults to true.
detect_table_columns  boolean optional. If true, detect the table columns in the file including the sql types. If false, skip table column detection. Defaults to false.
force_character_set_conversion  boolean optional. If true, the file will always be converted to UTF-8 and any character that cannot be converted will be discarded. If false, the character set conversion will only run if the detected character set is not compatible with UTF-8 (e.g., UTF-8, ASCII).
include_header  boolean optional. A boolean value indicating whether or not the first row of the file is a header row. If not provided, will attempt to auto-detect whether a header row is present.
column_delimiter  string optional. The column delimiter for the file. One of "comma", "tab", or "pipe". If not provided, the column delimiter will be auto-detected.

Value

A list containing the following elements:
id  integer, The ID of the job created.
files_put_preprocess_csv_archive

Update the archive status of this object

Description
Update the archive status of this object

Usage
files_put_preprocess_csv_archive(id, status)

Arguments
id integer required. The ID of the object.
status boolean required. The desired archived status of the object.

Value
A list containing the following elements:
id integer, The ID of the job created.
fileId integer, The ID of the file.
inPlace boolean, If true, the file is cleaned in place. If false, a new file ID is created. Defaults to true.

detectTableColumns boolean, If true, detect the table columns in the file including the sql types. If false, skip table column detection. Defaults to false.

forceCharacterSetConversion boolean, If true, the file will always be converted to UTF-8 and any character that cannot be converted will be discarded. If false, the character set conversion will only run if the detected character set is not compatible with UTF-8 (e.g., UTF-8, ASCII).

includeHeader boolean, A boolean value indicating whether or not the first row of the file is a header row. If not provided, will attempt to auto-detect whether a header row is present.

columnDelimiter string, The column delimiter for the file. One of "comma", "tab", or "pipe". If not provided, the column delimiter will be auto-detected.

hidden boolean, The hidden status of the item.
files_put_projects

Add a File to a project

Description

Add a File to a project

Usage

files_put_projects(id, project_id)

Arguments

id

integer required. The ID of the File.

project_id

integer required. The ID of the project.

Value

An empty HTTP response
files_put_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

files_put_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)

Arguments

id
    integer required. The ID of the resource that is shared.

group_ids
    array required. An array of one or more group IDs.

permission_level
    string required. Options are: "read", "write", or "manage".

share_email_body
    string optional. Custom body text for e-mail sent on a share.

send_shared_email
    boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers
    list, A list containing the following elements:
        • users array,
        • groups array,

writers
    list, A list containing the following elements:
        • users array,
        • groups array,

owners
    list, A list containing the following elements:
        • users array,
        • groups array,

totalUserShares
    integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

files_put_shares_users
Set the permissions users have on this object

Description
Set the permissions users have on this object

Usage
files_put_shares_users(
id, 
user_ids, 
permission_level, 
share_email_body = NULL, 
send_shared_email = NULL
)

Arguments
id integer required. The ID of the resource that is shared.
user_ids array required. An array of one or more user IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value
A list containing the following elements:
readers list, A list containing the following elements:
  • users array,
  • groups array,
writers list, A list containing the following elements:
  • users array,
  • groups array,
owners list, A list containing the following elements:
  • users array,
generate_client

• groups array,

totalUserShares
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

generate_client  Generate a client

Description

Generate a client

Usage

generate_client(spec)

Arguments

spec usually from get_spec

Value

A string containing one documented function for each verb at each endpoint.

get_database_id  Return the database id for a given database name

Description

Return the database id for a given database name

Usage

get_database_id(database_name)

Arguments

database_name string, Name of database.

Value

database_id integer, id of the matching database.
get_error

Get error data from civis_errors

Description

Get error data from civis_errors

Usage

get_error(x)

Arguments

x The error, usually from tryCatch

Value

A list containing

f The function
args A list of named arguments f was called with
error The error message from platform (if any)
status The status

Examples

```r
## Not run:
q_id <- queries_post(db_id, query, n_rows, cred_id)["id"]
r <- tryCatch(await(queries_get, id = q_id), error = function(e) e)
print(r)
get_error(r)

r <- try(await(queries_get, id = q_id))
get_error(r)

## End(Not run)
```
get_feature_importance

Get model feature importance

Description
Get model feature importance

Usage
get_feature_importance(model)

Arguments
model Object from civis_ml

Value
A matrix of features and their importance, ranked in descending order of importance

get_job_type_version Derive the job type and version from the given alias.

Description
Derive the job type and version from the given alias.

Usage
get_job_type_version(alias)

Arguments
alias A one-length character vector of the CivisML alias

Value
A list containing the job_type and version
get_metric

Get model metrics

Description

Get model metrics

Usage

get_metric(model, name = NULL)

Arguments

model Object from civis_ml
name Name of the metric. If NULL, all metrics are returned. Possible metrics are listed in details.

Details

The list of possible metrics provided by CivisML is:
Classification:
Regression:
"mad", "rmse", "r_squared", "score_histogram", "training_histogram", "y_yhat_plot", "y_yhat_outlier_rows", "oos_score_table"

Value

The metric given by name, or all metrics.

generate_response
get_status

Arguments

job output of jobs_get
verb one of "list" or "post"
fun_type one of "logs" or "outputs"

Details

container and custom scripts both have the same job type, but can be distinguished by a non-null fromTemplateId.

Value

The correct output or log fetching function (e.g. scripts_list_containers_runs_logs) based on the job type.

get_status Get the status from results of await

Description

Get the status from results of await

Usage

get_status(response)

Arguments

response the results from await

See Also

await await_all
get_table_id  
Return the table id for a given table

Description
Return the table id for a given table

Usage
get_table_id(table_name, database = NULL)

Arguments
- table_name: string, Name of table and schema "schema.tablename"
- database: string, Name of database where data frame is to be uploaded. If no database is specified, uses options(civis.default_db)

Value
table_id: integer, id of the matching table.

See Also
Other tables: refresh_table(), transfer_table()

generate_ids_all_versions
Get template IDs for all accessible CivisML versions.

Description
Get template IDs for all accessible CivisML versions.

Usage
generate_ids_all_versions()

Value
A data frame containing the template id, CivisML version, and job type.
get_train_template_id

Get training template IDs for the specified CivisML version

Description
Get training template IDs for the specified CivisML version

Usage
get_train_template_id(civisml_version = "prod")

Arguments

  civisml_version
      A one-length character vector of the CivisML version

Value
The template ID for training

git_repos_delete
Remove the bookmark on a git repository

Description
Remove the bookmark on a git repository

Usage
git_repos_delete(id)

Arguments

  id
      integer required. The ID for this git repository.

Value
An empty HTTP response
### git_repos_get

**Get a bookmarked git repository**

**Description**
Get a bookmarked git repository

**Usage**

```plaintext
git_repos_get(id)
```

**Arguments**

- `id` integer required. The ID for this git repository.

**Value**
A list containing the following elements:

- `id` integer, The ID for this git repository.
- `repoUrl` string, The URL for this git repository.
- `createdAt` string,
- `updatedAt` string,

### git_repos_list

**List bookmarked git repositories**

**Description**
List bookmarked git repositories

**Usage**

```plaintext
git_repos_list(limit = NULL, page_num = NULL, order = NULL, order_dir = NULL)
```

**Arguments**

- `limit` integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
- `page_num` integer optional. Page number of the results to return. Defaults to the first page, 1.
- `order` string optional. The field on which to order the result set. Defaults to repo_url. Must be one of: repo_url, created_at.
- `order_dir` string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.
Value

An array containing the following fields:

- id: integer, The ID for this git repository.
- repoUrl: string, The URL for this git repository.
- createdAt: string,
- updatedAt: string,

---

**git_repos_post**  
*Bookmark a git repository*

Description

Bookmark a git repository

Usage

```
git_repos_post(repo_url)
```

Arguments

- repo_url: string required. The URL for this git repository.

Value

A list containing the following elements:

- id: integer, The ID for this git repository.
- repoUrl: string, The URL for this git repository.
- createdAt: string,
- updatedAt: string,
groups_delete

Delete a Group (deprecated)

Description
Delete a Group (deprecated)

Usage

groups_delete(id)

Arguments

id  integer required.

Value

An empty HTTP response

groups_delete_members

Remove a user from a group

Description
Remove a user from a group

Usage

groups_delete_members(id, user_id)

Arguments

id  integer required. The ID of the group.
user_id  integer required. The ID of the user.

Value

An empty HTTP response
groups_delete_shares_groups

Revoke the permissions a group has on this object

Description
Revoke the permissions a group has on this object

Usage

groups_delete_shares_groups(id, group_id)

Arguments

id integer required. The ID of the resource that is shared.
group_id integer required. The ID of the group.

Value
An empty HTTP response

groups_delete_shares_users

Revoke the permissions a user has on this object

Description
Revoke the permissions a user has on this object

Usage

groups_delete_shares_users(id, user_id)

Arguments

id integer required. The ID of the resource that is shared.
user_id integer required. The ID of the user.

Value
An empty HTTP response
get a Group

Usage

groups_get(id)

Arguments

id integer required.

Value

A list containing the following elements:

id integer, The ID of this group.

name string, This group’s name.

createdAt string, The date and time when this group was created.

description string, The description of the group.

slug string, The slug for this group.

organizationId integer, The ID of the organization this group belongs to.

organizationName string, The name of the organization this group belongs to.

memberCount integer, The total number of members in this group.

mustAgreeToEula boolean, Whether or not members of this group must sign the EULA.

defaultOtpRequiredForLogin boolean, The two factor authentication requirement for this group.

roleIds array, An array of ids of all the roles this group has.

defaultTimeZone string, The default time zone of this group.

defaultJobsLabel string, The default partition label for jobs of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.

defaultNotebooksLabel string, The default partition label for notebooks of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.
**groups_list**

**defaultServicesLabel**

string. The default partition label for services of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.

**members**

array, An array containing the following fields:

- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

---

**groups_list**

**List Groups**

**Description**

List Groups

**Usage**

```python
groups_list(
    query = NULL,
    permission = NULL,
    include_members = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```

**Arguments**

- **query** string optional. If specified, it will filter the groups returned. Infix matching is supported (e.g., "query=group" will return "group" and "group of people" and "my group" and "my group of people").
- **permission** string optional. A permissions string, one of "read", "write", or "manage". Lists only groups for which the current user has that permission.
- **include_members** boolean optional. Show members of the group.
- **limit** integer optional. Number of results to return. Defaults to 50. Maximum allowed is 1000.
- **page_num** integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order** string optional. The field on which to order the result set. Defaults to name. Must be one of: name, created_at.
- **order_dir** string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.
Value

An array containing the following fields:

- **id** integer, The ID of this group.
- **name** string, This group’s name.
- **createdAt** string, The date and time when this group was created.
- **slug** string, The slug for this group.
- **organizationId** integer, The ID of the organization this group belongs to.
- **organizationName** string, The name of the organization this group belongs to.
- **memberCount** integer, The total number of members in this group.
- **members** array, An array containing the following fields:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.

Description

List users and groups permissioned on this object

Usage

```ruby
groups_list_shares(id)
```

Arguments

- **id** integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

- **readers** list, A list containing the following elements:
  - **users** array,
  - **groups** array,

- **writers** list, A list containing the following elements:
  - **users** array,
  - **groups** array,
owners list, A list containing the following elements:

- users array,
- groups array.

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**groups_patch** Update some attributes of this Group

**Description**

Update some attributes of this Group

**Usage**

```ruby
groups_patch(
  id,
  name = NULL,
  description = NULL,
  slug = NULL,
  organization_id = NULL,
  must_agree_to_eula = NULL,
  default_otp_required_for_login = NULL,
  role_ids = NULL,
  default_time_zone = NULL,
  default_jobs_label = NULL,
  default_notebooks_label = NULL,
  default_services_label = NULL
)
```

**Arguments**

- **id** integer required. The ID of this group.
- **name** string optional. This group’s name.
- **description** string optional. The description of the group.
- **slug** string optional. The slug for this group.
- **organization_id** integer optional. The ID of the organization this group belongs to.
- **must_agree_to_eula** boolean optional. Whether or not members of this group must sign the EULA.
default_otp_required_for_login
  boolean optional. The two factor authentication requirement for this group.

role_ids
  array optional. An array of ids of all the roles this group has.

default_time_zone
  string optional. The default time zone of this group.

default_jobs_label
  string optional. The default partition label for jobs of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.

default_notebooks_label
  string optional. The default partition label for notebooks of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.

default_services_label
  string optional. The default partition label for services of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.

Value

A list containing the following elements:

id
  integer, The ID of this group.

name
  string, This group’s name.

createdAt
  string, The date and time when this group was created.

description
  string, The description of the group.

slug
  string, The slug for this group.

organizationId
  integer, The ID of the organization this group belongs to.

organizationName
  string, The name of the organization this group belongs to.

memberCount
  integer, The total number of members in this group.

mustAgreeToEula
  boolean, Whether or not members of this group must sign the EULA.

defaultOtpRequiredForLogin
  boolean, The two factor authentication requirement for this group.

roleIdIds
  array, An array of ids of all the roles this group has.

defaultTimeZone
  string, The default time zone of this group.

defaultJobsLabel
  string, The default partition label for jobs of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.

defaultNotebooksLabel
  string, The default partition label for notebooks of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.
defaultServicesLabel
string. The default partition label for services of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.

members array. An array containing the following fields:
  • id integer. The ID of this user.
  • name string. This user’s name.
  • username string. This user’s username.
  • initials string. This user’s initials.
  • online boolean. Whether this user is online.

---

Description
Create a Group

Usage

```ruby
groups_post(
  name,
  description = NULL,
  slug = NULL,
  organization_id = NULL,
  must_agree_to_eula = NULL,
  default_otp_required_for_login = NULL,
  role_ids = NULL,
  default_time_zone = NULL,
  default_jobs_label = NULL,
  default_notebooks_label = NULL,
  default_services_label = NULL
)
```

Arguments

name string required. This group’s name.
description string optional. The description of the group.
slug string optional. The slug for this group.
organization_id integer optional. The ID of the organization this group belongs to.
must_agree_to_eula boolean optional. Whether or not members of this group must sign the EULA.
default_otp_required_for_login boolean optional. The two factor authentication requirement for this group.
role_ids array optional. An array of ids of all the roles this group has.
default_time_zone string optional. The default time zone of this group.
default_jobs_label string optional. The default partition label for jobs of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.
default_notebooks_label string optional. The default partition label for notebooks of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.
default_services_label string optional. The default partition label for services of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.

Value

A list containing the following elements:
id integer, The ID of this group.
name string, This group's name.
createdAt string, The date and time when this group was created.
description string, The description of the group.
slug string, The slug for this group.
organizationId integer, The ID of the organization this group belongs to.
organizationName string, The name of the organization this group belongs to.
memberCount integer, The total number of members in this group.
mustAgreeToEula boolean, Whether or not members of this group must sign the EULA.
defaultOtpRequiredForLogin boolean, The two factor authentication requirement for this group.
roleIds array, An array of ids of all the roles this group has.
defaultTimeZone string, The default time zone of this group.
defaultJobsLabel string, The default partition label for jobs of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.
defaultNotebooksLabel string, The default partition label for notebooks of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.
defaultServicesLabel

string. The default partition label for services of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.

members

array. An array containing the following fields:

• id integer. The ID of this user.
• name string. This user’s name.
• username string. This user’s username.
• initials string. This user’s initials.
• online boolean. Whether this user is online.

---

Description

Replace all attributes of this Group

Usage

```
groups_put(
    id,
    name,
    description = NULL,
    slug = NULL,
    organization_id = NULL,
    must_agree_to_eula = NULL,
    default_otp_required_for_login = NULL,
    role_ids = NULL,
    default_time_zone = NULL,
    default_jobs_label = NULL,
    default_notebooks_label = NULL,
    default_services_label = NULL
)
```

Arguments

- id integer required. The ID of this group.
- name string required. This group’s name.
- description string optional. The description of the group.
- slug string optional. The slug for this group.
- organization_id integer optional. The ID of the organization this group belongs to.
- must_agree_to_eula boolean optional. Whether or not members of this group must sign the EULA.
default_otp_required_for_login
   boolean optional. The two factor authentication requirement for this group.

role_ids
   array optional. An array of ids of all the roles this group has.

default_time_zone
   string optional. The default time zone of this group.

default_jobs_label
   string optional. The default partition label for jobs of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.

default_notebooks_label
   string optional. The default partition label for notebooks of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.

default_services_label
   string optional. The default partition label for services of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.

Value

A list containing the following elements:

id
   integer, The ID of this group.

name
   string, This group’s name.

createdAt
   string, The date and time when this group was created.

description
   string, The description of the group.

slug
   string, The slug for this group.

organizationId
   integer, The ID of the organization this group belongs to.

organizationName
   string, The name of the organization this group belongs to.

memberCount
   integer, The total number of members in this group.

mustAgreeToEula
   boolean, Whether or not members of this group must sign the EULA.

defaultOtpRequiredForLogin
   boolean, The two factor authentication requirement for this group.

roleIds
   array, An array of ids of all the roles this group has.

defaultTimeZone
   string, The default time zone of this group.

defaultJobsLabel
   string, The default partition label for jobs of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.

defaultNotebooksLabel
   string, The default partition label for notebooks of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.
groups_put_members

defaultServicesLabel
string, The default partition label for services of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.

members array, An array containing the following fields:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

Description

Add a user to a group

Usage

groups_put_members(id, user_id)

Arguments

id integer required. The ID of the group.
user_id integer required. The ID of the user.

Value

A list containing the following elements:

id integer, The ID of this group.
name string, This group’s name.
createdAt string, The date and time when this group was created.
description string, The description of the group.
slug string, The slug for this group.
organizationId integer, The ID of the organization this group belongs to.
organizationName string, The name of the organization this group belongs to.
memberCount integer, The total number of members in this group.
mustAgreeToEula boolean, Whether or not members of this group must sign the EULA.
defaultOtpRequiredForLogin boolean, The two factor authentication requirement for this group.
roleIds array, An array of ids of all the roles this group has.
defaultTimeZone string, The default time zone of this group.
defaultJobsLabel string, The default partition label for jobs of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.
defaultNotebooksLabel string, The default partition label for notebooks of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.
defaultServicesLabel string, The default partition label for services of this group. Only available if custom_partitions feature flag is set. Do not use this attribute as it may break in the future.
members array, An array containing the following fields:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

---

groups_put_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```python
groups_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of the resource that is shared.</td>
</tr>
<tr>
<td>group_ids</td>
<td>array required. An array of one or more group IDs.</td>
</tr>
</tbody>
</table>
groups_put_shares_users  

| permission_level | string required. Options are: "read", "write", or "manage". |
| share_email_body | string optional. Custom body text for e-mail sent on a share. |
| send_shared_email | boolean optional. Send email to the recipients of a share. |

Value

A list containing the following elements:

readers list, A list containing the following elements:
  • users array,
  • groups array,

writers list, A list containing the following elements:
  • users array,
  • groups array,

owners list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

Description

Set the permissions users have on this object

Usage

```python
groups_put_shares_users(  
id,  
user_ids,  
permission_level,  
share_email_body = NULL,  
send_shared_email = NULL  
)```
Arguments

id integer required. The ID of the resource that is shared.
user_ids array required. An array of one or more user IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers list, A list containing the following elements:
  • users array,
  • groups array,
writers list, A list containing the following elements:
  • users array,
  • groups array,
owners list, A list containing the following elements:
  • users array,
  • groups array,
totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

hist.civis_ml

Plot a histogram of the predicted scores or probabilities for a CivisML object

Description

Plot a histogram of the predicted scores or probabilities for a CivisML object

Usage

```r
## S3 method for class 'civis_ml'
hist(x, name = NULL, ...)
```
imports_delete_files_csv

Arguments

- **x**: civis_ml object
- **name**: name of the class (for multiclass output)
- **...**: unused.

Value

A ggplot2 plot object is returned invisibly.

---

**imports_delete_files_csv**

*Archive a CSV Import (deprecated, use archiving endpoints instead)*

**Description**

Archive a CSV Import (deprecated, use archiving endpoints instead)

**Usage**

`imports_delete_files_csv(id)`

**Arguments**

- **id**: integer required.

**Value**

An empty HTTP response

---

**imports_delete_files_runs**

*Cancel a run*

**Description**

Cancel a run

**Usage**

`imports_delete_files_runs(id, run_id)`

**Arguments**

- **id**: integer required. The ID of the import.
- **run_id**: integer required. The ID of the run.
Value
An empty HTTP response

---

**imports_delete_projects**

*Remove an Import from a project*

Description
Remove an Import from a project

Usage
imports_delete_projects(id, project_id)

Arguments
- id: integer required. The ID of the Import.
- project_id: integer required. The ID of the project.

Value
An empty HTTP response

---

**imports_delete_shares_groups**

*Revoke the permissions a group has on this object*

Description
Revoke the permissions a group has on this object

Usage
imports_delete_shares_groups(id, group_id)

Arguments
- id: integer required. The ID of the resource that is shared.
- group_id: integer required. The ID of the group.

Value
An empty HTTP response
**Description**

Revoke the permissions a user has on this object

**Usage**

`imports_delete_shares_users(id, user_id)`

**Arguments**

- `id` integer required. The ID of the resource that is shared.
- `user_id` integer required. The ID of the user.

**Value**

An empty HTTP response

---

**Description**

Archive a sync (deprecated, use the /archive endpoint instead)

**Usage**

`imports_delete_syncs(id, sync_id)`

**Arguments**

- `id` integer required. The ID of the import to fetch.
- `sync_id` integer required. The ID of the sync to fetch.

**Value**

An empty HTTP response
**Description**

Get details about an import

**Usage**

`imports_get(id)`

**Arguments**

`id` integer required. The ID for the import.

**Value**

A list containing the following elements:

- `name` string, The name of the import.
- `syncType` string, The type of sync to perform; one of Dbsync, AutoImport, GdocImport, GdocExport, and Salesforce.
- `source` list, A list containing the following elements:
  - `remoteHostId` integer,
  - `credentialId` integer,
  - `additionalCredentials` array, Array that holds additional credentials used for specific imports. For salesforce imports, the first and only element is the client credential id. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.
  - `name` string,
- `destination` list, A list containing the following elements:
  - `remoteHostId` integer,
  - `credentialId` integer,
  - `additionalCredentials` array, Array that holds additional credentials used for specific imports. For salesforce imports, the first and only element is the client credential id. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.
  - `name` string,
- `schedule` list, A list containing the following elements:
  - `scheduled` boolean, If the item is scheduled.
  - `scheduledDays` array, Day based on numeric value starting at 0 for Sunday.
  - `scheduledHours` array, Hours of the day it is scheduled on.
• scheduledMinutes array, Minutes of the day it is scheduled on.
• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
• urls array, URLs to receive a POST request at job completion
• successEmailSubject string, Custom subject line for success e-mail.
• successEmailBody string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

parentId integer, Parent id to trigger this import from
id integer, The ID for the import.
isOutbound boolean,
jobType string, The job type of this import.
syncs array, An array containing the following fields:
• id integer,
• source object,
• destination object,
• advancedOptions object,

state string,
createdAt string,
updatedAt string,
lastRun list, A list containing the following elements:
• id integer,
• state string,
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.
• error string, The error message for this run, if present.

user list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.
timeZone string, The time zone of this import.
hidden boolean, The hidden status of the item.
archived string, The archival status of the requested item(s).

---

**imports_get_batches**

*Get details about a batch import*

**Description**

Get details about a batch import

**Usage**

`imports_get_batches(id)`

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID for the import.</td>
</tr>
</tbody>
</table>

**Value**

A list containing the following elements:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer, The ID for the import.</td>
</tr>
<tr>
<td>schema</td>
<td>string, The destination schema name. This schema must already exist in Redshift.</td>
</tr>
<tr>
<td>table</td>
<td>string, The destination table name, without the schema prefix. This table must already exist in Redshift.</td>
</tr>
<tr>
<td>remoteHostId</td>
<td>integer, The ID of the destination database host.</td>
</tr>
<tr>
<td>state</td>
<td>string, The state of the run; one of &quot;queued&quot;, &quot;running&quot;, &quot;succeeded&quot;, &quot;failed&quot;, or &quot;cancelled&quot;.</td>
</tr>
<tr>
<td>startedAt</td>
<td>string, The time the last run started at.</td>
</tr>
<tr>
<td>finishedAt</td>
<td>string, The time the last run completed.</td>
</tr>
<tr>
<td>error</td>
<td>string, The error returned by the run, if any.</td>
</tr>
<tr>
<td>hidden</td>
<td>boolean, The hidden status of the item.</td>
</tr>
</tbody>
</table>
Get a CSV Import

**Usage**

`imports_get_files_csv(id)`

**Arguments**

- **id**
  - integer required.

**Value**

A list containing the following elements:

- **id**
  - integer, The ID for the import.
- **name**
  - string, The name of the import.
- **source**
  - list, A list containing the following elements:
    - **fileIds** array, The file ID(s) to import, if importing Civis file(s).
    - **storagePath** list. A list containing the following elements:
      - **storageHostId** integer, The ID of the source storage host.
      - **credentialId** integer, The ID of the credentials for the source storage host.
      - **filePaths** array, The file or directory path(s) within the bucket from which to import. E.g. the file path for "s3://mybucket/files/all/" would be "/files/all/"If specifying a directory path, the job will import every file found under that path. All files must have the same column layout and file format (e.g., compression, columnDelimiter, etc.).
- **destination**
  - list, A list containing the following elements:
    - **schema** string, The destination schema name.
    - **table** string, The destination table name.
    - **remoteHostId** integer, The ID of the destination database host.
    - **credentialId** integer, The ID of the credentials for the destination database.
    - **primaryKeys** array, A list of column(s) which together uniquely identify a row in the destination table. These columns must not contain NULL values. If the import mode is "upsert", this field is required; see the Civis Helpdesk article on "Advanced CSV Imports via the Civis API" for more information.
    - **lastModifiedKeys** array, A list of the columns indicating a record has been updated. If the destination table does not exist, and the import mode is "upsert", this field is required.
firstRowIsHeader
    boolean, A boolean value indicating whether or not the first row of the source file is a header row.

columnDelimiter
    string, The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

escaped
    boolean, A boolean value indicating whether or not the source file has quotes escaped with a backslash. Defaults to false.

compression
    string, The type of compression of the source file. Valid arguments are "gzip" and "none". Defaults to "none".

existingTableRows
    string, The behavior if a destination table with the requested name already exists. One of "fail", "truncate", "append", "drop", or "upsert". Defaults to "fail".

maxErrors
    integer, The maximum number of rows with errors to ignore before failing. This option is not supported for Postgres databases.

tableColumns
    array, An array containing the following fields:
    - name string, The column name.
    - sqlType string, The SQL type of the column.

loosenTypes
    boolean, If true, SQL types with precisions/lengths will have these values increased to accommodate data growth in future loads. Type loosening only occurs on table creation. Defaults to false.

execution
    string, In upsert mode, controls the movement of data in upsert mode. If set to "delayed", the data will be moved after a brief delay. If set to "immediate", the data will be moved immediately. In non-upsert modes, controls the speed at which detailed column stats appear in the data catalogue. Defaults to "delayed", to accommodate concurrent upserts to the same table and speedier non-upsert imports.

redshiftDestinationOptions
    list, A list containing the following elements:
    - diststyle string, The diststyle to use for the table. One of "even", "all", or "key".
    - distkey string, Distkey for this table in Redshift
    - sortkeys array, Sortkeys for this table in Redshift. Please provide a maximum of two.

hidden
    boolean, The hidden status of the item.

---

**Imports Get Files Runs**

*Check status of a run*

**Description**

Check status of a run
Usage

`imports_get_files_runs(id, run_id)`

Arguments

id  integer required. The ID of the import.
run_id  integer required. The ID of the run.

Value

A list containing the following elements:

id  integer, The ID of the run.
importId  integer, The ID of the import.
state  string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested  boolean, True if run cancel requested, else false.
startedAt  string, The time the last run started at.
finishedAt  string, The time the last run completed.
error  string, The error, if any, returned by the run.

Description

List Imports

Usage

```python
imports_list(
    type = NULL,
    author = NULL,
    destination = NULL,
    source = NULL,
    status = NULL,
    hidden = NULL,
    archived = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```
**Arguments**

- **type**
  - string optional. If specified, return imports of these types. It accepts a comma-separated list, possible values are 'AutoImport', 'DbSync', 'Salesforce', 'GdocImport'.

- **author**
  - string optional. If specified, return imports from this author. It accepts a comma-separated list of author ids.

- **destination**
  - string optional. If specified, returns imports with one of these destinations. It accepts a comma-separated list of remote host ids.

- **source**
  - string optional. If specified, returns imports with one of these sources. It accepts a comma-separated list of remote host ids. 'DbSync' must be specified for 'type'.

- **status**
  - string optional. If specified, returns imports with one of these statuses. It accepts a comma-separated list, possible values are 'running', 'failed', 'succeeded', 'idle', 'scheduled'.

- **hidden**
  - boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

- **archived**
  - string optional. The archival status of the requested item(s).

- **limit**
  - integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.

- **page_num**
  - integer optional. Page number of the results to return. Defaults to the first page, 1.

- **order**
  - string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at, last_run.updated_at.

- **order_dir**
  - string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

**Value**

An array containing the following fields:

- **name**
  - string, The name of the import.

- **syncType**
  - string, The type of sync to perform; one of Dbsync, AutoImport, GdocImport, GdocExport, and Salesforce.

- **source**
  - list, A list containing the following elements:
    - remoteHostId integer,
    - credentialId integer,
    - additionalCredentials array, Array that holds additional credentials used for specific imports. For salesforce imports, the first and only element is the client credential id. For DB Synacs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.
    - name string,

- **destination**
  - list, A list containing the following elements:
    - remoteHostId integer,
- credentialId integer,
- additionalCredentials array, Array that holds additional credentials used for specific imports. For salesforce imports, the first and only element is the client credential id. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.
- name string,

schedule list, A list containing the following elements:
- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

id integer, The ID for the import.

isOutbound boolean,

jobType string, The job type of this import.

state string,

createdAt string,

updatedAt string,

lastRun list, A list containing the following elements:
- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

user list, A list containing the following elements:
- id integer, The ID of this user.
- name string, This user’s name.
- username string, This user’s username.
- initials string, This user’s initials.
- online boolean, Whether this user is online.

timeZone string, The time zone of this import.

archived string, The archival status of the requested item(s).
**imports_list_batches**

*List batch imports*

**Description**

List batch imports

**Usage**

```r
imports_list_batches(
  hidden = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

**Arguments**

- `hidden` boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
- `limit` integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
- `page_num` integer optional. Page number of the results to return. Defaults to the first page, 1.
- `order` string optional. The field on which to order the result set. Defaults to `updated_at`. Must be one of: `updated_at`, `created_at`.
- `order_dir` string optional. Direction in which to sort, either `asc` (ascending) or `desc` (descending) defaulting to `desc`.

**Value**

An array containing the following fields:

- `id` integer, The ID for the import.
- `schema` string, The destination schema name. This schema must already exist in Redshift.
- `table` string, The destination table name, without the schema prefix. This table must already exist in Redshift.
- `remoteHostId` integer, The ID of the destination database host.
- `state` string, The state of the run; one of "queued", "running", "succeeded", "failed", or "cancelled".
- `startedAt` string, The time the last run started at.
- `finishedAt` string, The time the last run completed.
- `error` string, The error returned by the run, if any.
imports_list_files_runs

List runs for the given import

Description

List runs for the given import

Usage

imports_list_files_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL)

Arguments

id integer required. The ID of the import.
limit integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num integer optional. Page number of the results to return. Defaults to the first page, 1.
order string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id integer, The ID of the run.
importId integer, The ID of the import.
state string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested boolean, True if run cancel requested, else false.
startedAt string, The time the last run started at.
finishedAt string, The time the last run completed.
error string, The error, if any, returned by the run.
imports_list_files_runs_logs

Get the logs for a run

Description
Get the logs for a run

Usage
imports_list_files_runs_logs(id, run_id, last_id = NULL, limit = NULL)

Arguments
id integer required. The ID of the import.
run_id integer required. The ID of the run.
last_id integer optional. The ID of the last log message received. Log entries with this
ID value or lower will be omitted. Logs are sorted by ID if this value is provided,
and are otherwise sorted by createdAt.
limit integer optional. The maximum number of log messages to return. Default of
10000.

Value
An array containing the following fields:
id integer, The ID of the log.
createdAt string, The time the log was created.
message string, The log message.
level string, The level of the log. One of unknown,fatal,error,warn,info,debug.

imports_list_projects
List the projects an Import belongs to

Description
List the projects an Import belongs to

Usage
imports_list_projects(id, hidden = NULL)
**imports_list_runs**

**Arguments**

- **id** integer required. The ID of the Import.
- **hidden** boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

**Value**

An array containing the following fields:

- **id** integer, The ID for this project.
- **author** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **name** string, The name of this project.
- **description** string, A description of the project.
- **users** array, An array containing the following fields:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **autoShare** boolean,
- **createdAt** string,
- **updatedAt** string,
- **archived** string, The archival status of the requested item(s).

---

**imports_list_runs** Get the run history of this import

---

**Description**

Get the run history of this import

**Usage**

`imports_list_runs(id)`

**Arguments**

- **id** integer required.
Value

An array containing the following fields:

- `id` integer,
- `state` string,
- `createdAt` string, The time that the run was queued.
- `startedAt` string, The time that the run started.
- `finishedAt` string, The time that the run completed.
- `error` string, The error message for this run, if present.

---

Get the logs for a run

Description

Get the logs for a run

Usage

`imports_list_runs_logs(id, run_id, last_id = NULL, limit = NULL)`

Arguments

- `id` integer required. The ID of the import.
- `run_id` integer required. The ID of the run.
- `last_id` integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted. Logs are sorted by ID if this value is provided, and are otherwise sorted by `createdAt`.
- `limit` integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

- `id` integer, The ID of the log.
- `createdAt` string, The time the log was created.
- `message` string, The log message.
- `level` string, The level of the log. One of unknown,fatal,error,warn,info,debug.
Description

List users and groups permissioned on this object

Usage

imports_list_shares(id)

Arguments

id    integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers  list, A list containing the following elements:
   • users array,
   • groups array,

writers  list, A list containing the following elements:
   • users array,
   • groups array,

owners   list, A list containing the following elements:
   • users array,
   • groups array,

totalUserShares
   integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
   integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.
Update some attributes of this CSV Import

Description

Update some attributes of this CSV Import

Usage

```r
imports_patch_files_csv(
  id,
  name = NULL,
  source = NULL,
  destination = NULL,
  first_row_is_header = NULL,
  column_delimiter = NULL,
  escaped = NULL,
  compression = NULL,
  existing_table_rows = NULL,
  max_errors = NULL,
  table_columns = NULL,
  loosen_types = NULL,
  execution = NULL,
  redshift_destination_options = NULL
)
```

Arguments

**id**
integer required. The ID for the import.

**name**
string optional. The name of the import.

**source**
list optional. A list containing the following elements:
- `fileIds` array, The file ID(s) to import, if importing Civis file(s).
- `storagePath` list. A list containing the following elements:
  - `storageHostId` integer, The ID of the source storage host.
  - `credentialId` integer, The ID of the credentials for the source storage host.
  - `filePaths` array, The file or directory path(s) within the bucket from which to import. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"If specifying a directory path, the job will import every file found under that path. All files must have the same column layout and file format (e.g., compression, columnDelimiter, etc.).

**destination**
list optional. A list containing the following elements:
- `schema` string, The destination schema name.
- `table` string, The destination table name.
• remoteHostId integer, The ID of the destination database host.
• credentialId integer, The ID of the credentials for the destination database.
• primaryKeys array, A list of column(s) which together uniquely identify a row in the destination table. These columns must not contain NULL values. If the import mode is "upsert", this field is required; see the Civis Helpdesk article on "Advanced CSV Imports via the Civis API" for more information.
• lastModifiedKeys array, A list of the columns indicating a record has been updated. If the destination table does not exist, and the import mode is "upsert", this field is required.

first_row_is_header
boolean optional. A boolean value indicating whether or not the first row of the source file is a header row.

column_delimiter
string optional. The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

escaped
boolean optional. A boolean value indicating whether or not the source file has quotes escaped with a backslash. Defaults to false.

compression
string optional. The type of compression of the source file. Valid arguments are "gzip" and "none". Defaults to "none".

existing_table_rows
string optional. The behavior if a destination table with the requested name already exists. One of "fail", "truncate", "append", "drop", or "upsert". Defaults to "fail".

max_errors
integer optional. The maximum number of rows with errors to ignore before failing. This option is not supported for Postgres databases.

table_columns
array optional. An array containing the following fields:
  • name string, The column name.
  • sqlType string, The SQL type of the column.

loosen_types
boolean optional. If true, SQL types with precisions/lengths will have these values increased to accommodate data growth in future loads. Type loosening only occurs on table creation. Defaults to false.

execution
string optional. In upsert mode, controls the movement of data in upsert mode. If set to "delayed", the data will be moved after a brief delay. If set to "immediate", the data will be moved immediately. In non-upsert modes, controls the speed at which detailed column stats appear in the data catalogue. Defaults to "delayed", to accommodate concurrent upserts to the same table and speedier non-upsert imports.

redshift_destination_options
list optional. A list containing the following elements:
  • diststyle string, The diststyle to use for the table. One of "even", "all", or "key".
  • distkey string, Distkey for this table in Redshift
  • sortkeys array, Sortkeys for this table in Redshift. Please provide a maximum of two.
Value

A list containing the following elements:

- id integer, The ID for the import.
- name string, The name of the import.
- source list, A list containing the following elements:
  - fileIds array, The file ID(s) to import, if importing Civis file(s).
  - storagePath list, A list containing the following elements:
    - credentialId integer, The ID of the credentials for the source storage host.
    - filePaths array, The file or directory path(s) within the bucket from which to import. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/" If specifying a directory path, the job will import every file found under that path. All files must have the same column layout and file format (e.g., compression, columnDelimiter, etc.).
- destination list, A list containing the following elements:
  - schema string, The destination schema name.
  - table string, The destination table name.
  - remoteHostId integer, The ID of the destination database host.
  - credentialId integer, The ID of the credentials for the destination database.
  - primaryKeys array, A list of column(s) which together uniquely identify a row in the destination table. These columns must not contain NULL values. If the import mode is "upsert", this field is required; see the Civis Helpdesk article on "Advanced CSV Imports via the Civis API" for more information.
  - lastModifiedKeys array, A list of the columns indicating a record has been updated. If the destination table does not exist, and the import mode is "upsert", this field is required.
- firstRowIsHeader boolean, A boolean value indicating whether or not the first row of the source file is a header row.
- columnDelimiter string, The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".
- escaped boolean, A boolean value indicating whether or not the source file has quotes escaped with a backslash. Defaults to false.
- compression string, The type of compression of the source file. Valid arguments are "gzip" and "none". Defaults to "none".
- existingTableRows string, The behavior if a destination table with the requested name already exists. One of "fail", "truncate", "append", "drop", or "upsert". Defaults to "fail".
- maxErrors integer, The maximum number of rows with errors to ignore before failing. This option is not supported for Postgres databases.
- tableColumns array, An array containing the following fields:
Create a new import configuration

**Description**

Create a new import configuration

**Usage**

```python
imports_post(
    name,
    sync_type,
    is_outbound,
    source = NULL,
    destination = NULL,
    schedule = NULL,
    notifications = NULL,
    parent_id = NULL,
    next_run_at = NULL,
    time_zone = NULL,
    hidden = NULL
)
```
**Arguments**

- **name** string required. The name of the import.
- **sync_type** string required. The type of sync to perform; one of Dbsync, AutoImport, GdocImport, GdocExport, and Salesforce.
- **is_outbound** boolean required.
- **source** list optional. A list containing the following elements:
  - remoteHostId integer,
  -credentialId integer,
  - additionalCredentials array, Array that holds additional credentials used for specific imports. For salesforce imports, the first and only element is the client credential id. For DB Syns, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.
- **destination** list optional. A list containing the following elements:
  - remoteHostId integer,
  - credentialId integer,
  - additionalCredentials array, Array that holds additional credentials used for specific imports. For salesforce imports, the first and only element is the client credential id. For DB Syns, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.
- **schedule** list optional. A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.
- **notifications** list optional. A list containing the following elements:
  - urls array, URLs to receive a POST request at job completion
  - successEmailSubject string, Custom subject line for success e-mail.
  - successEmailBody string, Custom body text for success e-mail, written in Markdown.
  - successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  - successEmailFromName string, Name from which success emails are sent; defaults to “Civis.”
  - successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  - failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  - stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

parent_id  integer optional. Parent id to trigger this import from
next_run_at  string optional. The time of the next scheduled run.
time_zone  string optional. The time zone of this import.
hidden  boolean optional. The hidden status of the item.

Value

A list containing the following elements:

ame  string, The name of the import.
syncType  string, The type of sync to perform; one of Dbsync, AutoImport, GdocImport, GdocExport, and Salesforce.
source  list, A list containing the following elements:
  • remoteHostId integer,
  • credentialId integer,
  • additionalCredentials array, Array that holds additional credentials used for specific imports. For salesforce imports, the first and only element is the client credential id. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.
  • name string.

destination  list, A list containing the following elements:
  • remoteHostId integer,
  • credentialId integer,
  • additionalCredentials array, Array that holds additional credentials used for specific imports. For salesforce imports, the first and only element is the client credential id. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.
  • name string.

schedule  list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications  list, A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

**parentId**
integer, Parent id to trigger this import from

**id**
integer, The ID for the import.

**isOutbound**
boolean,

**jobType**
string, The job type of this import.

**syncs**
array, An array containing the following fields:
  - id integer,
  - source object,
  - destination object,
  - advancedOptions object,

**state**
string,

**createdAt**
string,

**updatedAt**
string,

**lastRun**
list, A list containing the following elements:
  - id integer,
  - state string,
  - createdAt string, The time that the run was queued.
  - startedAt string, The time that the run started.
  - finishedAt string, The time that the run completed.
  - error string, The error message for this run, if present.

**user**
list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user's name.
  - username string, This user's username.
  - initials string, This user's initials.
  - online boolean, Whether this user is online.

**runningAs**
list, A list containing the following elements:
  - id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

timeZone string, The time zone of this import.
hidden boolean, The hidden status of the item.
archived string, The archival status of the requested item(s).

---

imports_post_batches (Upload multiple files to Civis)

**Description**

Upload multiple files to Civis

**Usage**

```python
imports_post_batches(
    file_ids,
    schema,
    table,
    remote_host_id,
    credential_id,
    column_delimiter = NULL,
    first_row_is_header = NULL,
    compression = NULL,
    hidden = NULL
)
```

**Arguments**

- **file_ids** array required. The file IDs for the import.
- **schema** string required. The destination schema name. This schema must already exist in Redshift.
- **table** string required. The destination table name, without the schema prefix. This table must already exist in Redshift.
- **remote_host_id** integer required. The ID of the destination database host.
- **credential_id** integer required. The ID of the credentials to be used when performing the database import.
- **column_delimiter** string optional. The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". If unspecified, defaults to "comma".
first_row_is_header

  boolean optional. A boolean value indicating whether or not the first row is a header row. If unspecified, defaults to false.

compression

  string optional. The type of compression. Valid arguments are "gzip", "zip", and "none". If unspecified, defaults to "gzip".

hidden

  boolean optional. The hidden status of the item.

Value

  A list containing the following elements:

id

  integer, The ID for the import.

schema

  string, The destination schema name. This schema must already exist in Redshift.

table

  string, The destination table name, without the schema prefix. This table must already exist in Redshift.

remoteHostId

  integer, The ID of the destination database host.

state

  string, The state of the run; one of "queued", "running", "succeeded", "failed", or "cancelled".

startedAt

  string, The time the last run started at.

finishedAt

  string, The time the last run completed.

error

  string, The error returned by the run, if any.

hidden

  boolean, The hidden status of the item.

---

**imports_post_cancel  Cancel a run**

---

**Description**

  Cancel a run

**Usage**

  imports_post_cancel(id)

**Arguments**

id

  integer required. The ID of the job.

**Value**

  A list containing the following elements:

id

  integer, The ID of the run.

state

  string, The state of the run, one of 'queued', 'running' or 'cancelled'.

isCancelRequested

  boolean, True if run cancel requested, else false.
Initiate an import of a tabular file into the platform

**Usage**

```python
imports_post_files(
    schema,  # string required. The schema of the destination table.
    name,    # string required. The name of the destination table.
    remote_host_id,  # integer required. The id of the destination database host.
    credential_id,  # integer required. The id of the credentials to be used when performing the database import.
    max_errors = NULL,  # integer optional. The maximum number of rows with errors to remove from the import before failing.
    existing_table_rows = NULL,  # string optional. The behaviour if a table with the requested name already exists. One of "fail", "truncate", "append", or "drop". Defaults to "fail".
    diststyle = NULL,  # string optional. The diststyle to use for the table. One of "even", "all", or "key".
    distkey = NULL,  # string optional. The column to use as the distkey for the table.
    sortkey1 = NULL,  # string optional. The column to use as the sort key for the table.
    sortkey2 = NULL,  # string optional. The second column in a compound sortkey for the table.
    column_delimiter = NULL,  # string optional. The column delimiter for the table.
    first_row_is_header = NULL,  # string optional. The first row of the table.
    multipart = NULL,  # string optional. The multipart of the table.
    escaped = NULL,  # string optional. The escaped of the table.
    hidden = NULL
)
```

**Arguments**

- `schema`: string required. The schema of the destination table.
- `name`: string required. The name of the destination table.
- `remote_host_id`: integer required. The id of the destination database host.
- `credential_id`: integer required. The id of the credentials to be used when performing the database import.
- `max_errors`: integer optional. The maximum number of rows with errors to remove from the import before failing.
- `existing_table_rows`: string optional. The behaviour if a table with the requested name already exists. One of "fail", "truncate", "append", or "drop". Defaults to "fail".
- `diststyle`: string optional. The diststyle to use for the table. One of "even", "all", or "key".
- `distkey`: string optional. The column to use as the distkey for the table.
- `sortkey1`: string optional. The column to use as the sort key for the table.
- `sortkey2`: string optional. The second column in a compound sortkey for the table.
- `column_delimiter`: string optional. The column delimiter for the table.
- `first_row_is_header`: string optional. The first row of the table.
- `multipart`: string optional. The multipart of the table.
- `escaped`: string optional. The escaped of the table.
- `hidden`: string optional.
column_delimiter

String optional. The column delimiter of the file. If column_delimiter is null or omitted, it will be auto-detected. Valid arguments are "comma", "tab", and "pipe".

first_row_is_header

Boolean optional. A boolean value indicating whether or not the first row is a header row. If first_row_is_header is null or omitted, it will be auto-detected.

multipart

Boolean optional. If true, the upload URI will require a 'multipart/form-data' POST request. Defaults to false.

escaped

Boolean optional. If true, escape quotes with a backslash; otherwise, escape quotes by double-quoting. Defaults to false.

hidden

Boolean optional. The hidden status of the item.

Value

A list containing the following elements:

id

Integer, The id of the import.

uploadUri

String, The URI which may be used to upload a tabular file for import. You must use this URI to upload the file you wish imported and then inform the Civis API when your upload is complete using the URI given by the runUri field of this response.

runUri

String, The URI to POST to once the file upload is complete. After uploading the file using the URI given in the uploadUri attribute of the response, POST to this URI to initiate the import of your uploaded file into the platform.

uploadFields

List, If multipart was set to true, these fields should be included in the multipart upload.

Create a CSV Import

Description

Create a CSV Import

Usage

imports_post_files_csv(
    source,
    destination,
    first_row_is_header,
    name = NULL,
    column_delimiter = NULL,
    escaped = NULL,
    compression = NULL,
existing_table_rows = NULL,
max_errors = NULL,
table_columns = NULL,
loosen_types = NULL,
execution = NULL,
redshift_destination_options = NULL,
hidden = NULL
)

Arguments

source list required. A list containing the following elements:
• fileIds array, The file ID(s) to import, if importing Civis file(s).
• storagePath list. A list containing the following elements:
  – storageHostId integer, The ID of the source storage host.
  – credentialId integer, The ID of the credentials for the source storage host.
  – filePaths array, The file or directory path(s) within the bucket from which to import. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"If specifying a directory path, the job will import every file found under that path. All files must have the same column layout and file format (e.g., compression, columnDelimiter, etc.).

destination list required. A list containing the following elements:
• schema string, The destination schema name.
• table string, The destination table name.
• remoteHostId integer, The ID of the destination database host.
• credentialId integer, The ID of the credentials for the destination database.
• primaryKeys array, A list of column(s) which together uniquely identify a row in the destination table. These columns must not contain NULL values. If the import mode is "upsert", this field is required; see the Civis Helpdesk article on "Advanced CSV Imports via the Civis API" for more information.
• lastModifiedKeys array, A list of the columns indicating a record has been updated. If the destination table does not exist, and the import mode is "upsert", this field is required.

first_row_is_header boolean required. A boolean value indicating whether or not the first row of the source file is a header row.

name string optional. The name of the import.

column_delimiter string optional. The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

escaped boolean optional. A boolean value indicating whether or not the source file has quotes escaped with a backslash. Defaults to false.

compression string optional. The type of compression of the source file. Valid arguments are "gzip" and "none". Defaults to "none".
existing_table_rows
string optional. The behavior if a destination table with the requested name already exists. One of "fail", "truncate", "append", "drop", or "upsert". Defaults to "fail".

max_errors
integer optional. The maximum number of rows with errors to ignore before failing. This option is not supported for Postgres databases.

table_columns
array optional. An array containing the following fields:
• name string, The column name.
• sqlType string, The SQL type of the column.

loosen_types
boolean optional. If true, SQL types with precisions/lengths will have these values increased to accommodate data growth in future loads. Type loosening only occurs on table creation. Defaults to false.

execution
string optional. In upsert mode, controls the movement of data in upsert mode. If set to "delayed", the data will be moved after a brief delay. If set to "immediate", the data will be moved immediately. In non-upsert modes, controls the speed at which detailed column stats appear in the data catalogue. Defaults to "delayed", to accommodate concurrent upserts to the same table and speedier non-upsert imports.

redshift_destination_options
list optional. A list containing the following elements:
• diststyle string, The diststyle to use for the table. One of "even", "all", or "key".
• distkey string, Distkey for this table in Redshift
• sortkeys array, Sortkeys for this table in Redshift. Please provide a maximum of two.

hidden
boolean optional. The hidden status of the item.

Value
A list containing the following elements:

id
integer, The ID for the import.

name
string, The name of the import.

source
list, A list containing the following elements:
• fileIds array, The file ID(s) to import, if importing Civis file(s).
• storagePath list, A list containing the following elements:
  – storageHostId integer, The ID of the source storage host.
  – credentialId integer, The ID of the credentials for the source storage host.
  – filePaths array, The file or directory path(s) within the bucket from which to import. E.g. the file_path for "s3: //mybucket/files/all/" would be "/files/all/"If specifying a directory path, the job will import every file found under that path. All files must have the same column layout and file format (e.g., compression, columnDelimiter, etc.).

destination
list, A list containing the following elements:
• schema string, The destination schema name.
• table string, The destination table name.
• remoteHostId integer, The ID of the destination database host.
• credentialId integer, The ID of the credentials for the destination database.
• primaryKeys array, A list of column(s) which together uniquely identify a row in the destination table. These columns must not contain NULL values. If the import mode is "upsert", this field is required; see the Civis Helpdesk article on "Advanced CSV Imports via the Civis API" for more information.
• lastModifiedKeys array, A list of the columns indicating a record has been updated. If the destination table does not exist, and the import mode is "upsert", this field is required.

firstRowIsHeader

boolean, A boolean value indicating whether or not the first row of the source file is a header row.

columnDelimiter

string, The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

escaped

boolean, A boolean value indicating whether or not the source file has quotes escaped with a backslash. Defaults to false.

compression

string, The type of compression of the source file. Valid arguments are "gzip" and "none". Defaults to "none".

existingTableRows

string, The behavior if a destination table with the requested name already exists. One of "fail", "truncate", "append", "drop", or "upsert". Defaults to "fail".

maxErrors

integer, The maximum number of rows with errors to ignore before failing. This option is not supported for Postgres databases.

tableColumns

array, An array containing the following fields:
• name string, The column name.
• sqlType string, The SQL type of the column.

loosenTypes

boolean, If true, SQL types with precisions/lengths will have these values increased to accommodate data growth in future loads. Type loosening only occurs on table creation. Defaults to false.

exection

string, In upsert mode, controls the movement of data in upsert mode. If set to "delayed", the data will be moved after a brief delay. If set to "immediate", the data will be moved immediately. In non-upsert modes, controls the speed at which detailed column stats appear in the data catalogue. Defaults to "delayed", to accommodate concurrent upserts to the same table and speedier non-upsert imports.

redshiftDestinationOptions

list, A list containing the following elements:
• diststyle string, The diststyle to use for the table. One of "even", "all", or "key".
• distkey string, Distkey for this table in Redshift
• sortkeys array, Sortkeys for this table in Redshift. Please provide a maximum of two.

hidden

boolean, The hidden status of the item.
### Start a run

**Description**
Start a run

**Usage**

```python
imports_post_files_runs(id)
```

**Arguments**

- `id` integer required. The ID of the import.

**Value**
A list containing the following elements:

- `id` integer, The ID of the run.
- `importId` integer, The ID of the import.
- `state` string, The state of the run, one of 'queued', 'running', 'succeeded', 'failed' or 'cancelled'.
- `isCancelRequested` boolean, True if run cancel requested, else false.
- `startedAt` string, The time the last run started at.
- `finishedAt` string, The time the last run completed.
- `error` string, The error, if any, returned by the run.

### Run an import

**Description**
Run an import

**Usage**

```python
imports_post_runs(id)
```

**Arguments**

- `id` integer required. The ID of the import to run.
**Value**

A list containing the following elements:

- `runId` integer. The ID of the new run triggered.

---

**Description**

Create a sync

**Usage**

`imports_post_syncs(id, source, destination, advanced_options = NULL)`

**Arguments**

- `id` integer required.

- `source` list required.

  A list containing the following elements:

  - `path` string. The path of the dataset to sync from; for a database source, `schema.tablename`. If you are doing a Google Sheet export, this can be blank. This is a legacy parameter, it is recommended you use one of the following: `databaseTable`, `file`, `googleWorksheet`, `salesforce`

  - `databaseTable` list. A list containing the following elements:

    - `schema` string. The database schema name.
    - `table` string. The database table name.
    - `useWithoutSchema` boolean. This attribute is no longer available; defaults to false but cannot be used.

  - `file` list.

  - `googleWorksheet` list. A list containing the following elements:

    - `spreadsheet` string. The spreadsheet document name.
    - `spreadsheetId` string. The spreadsheet document id.
    - `worksheet` string. The worksheet tab name.
    - `worksheetId` integer. The worksheet tab id.

  - `salesforce` list. A list containing the following elements:

    - `objectName` string. The Salesforce object name.

- `destination` list required.

  A list containing the following elements:

  - `path` string. The `schema.tablename` to sync to. If you are doing a Google Sheet export, this is the spreadsheet and sheet name separated by a period, i.e. if you have a spreadsheet named "MySpreadsheet" and a sheet called "Sheet1" this field would be "MySpreadsheet.Sheet1". This is a legacy parameter, it is recommended you use one of the following: `databaseTable`, `file`, `googleWorksheet`
• databaseTable list. A list containing the following elements:
  – schema string, The database schema name.
  – table string, The database table name.
  – useWithoutSchema boolean, This attribute is no longer available; defaults to false but cannot be used.
• googleWorksheet list. A list containing the following elements:
  – spreadsheet string, The spreadsheet document name.
  – spreadsheetId string, The spreadsheet document id.
  – worksheet string, The worksheet tab name.
  – worksheetId integer, The worksheet tab id.

advanced_options

list optional. A list containing the following elements:
• maxErrors integer,
• existingTableRows string,
• diststyle string,
• distkey string,
• sortkey1 string,
• sortkey2 string,
• columnDelimiter string,
• columnOverrides object, Hash used for overriding auto-detected names and types, with keys being the index of the column being overridden.
• escaped boolean, If true, escape quotes with a backslash; otherwise, escape quotes by double-quotings. Defaults to false.
• identityColumn string,
• rowChunkSize integer,
• wipeDestinationTable boolean,
• truncateLongLines boolean,
• invalidCharReplacement string,
• verifyTableRowCounts boolean,
• partitionColumnName string, This parameter is deprecated
• partitionSchemaName string, This parameter is deprecated
• partitionTableName string, This parameter is deprecated
• partitionTablePartitionColumnMinName string, This parameter is deprecated
• partitionTablePartitionColumnMaxName string, This parameter is deprecated
• lastModifiedColumn string,
• mysqlCatalogMatchesSchema boolean, This attribute is no longer available; defaults to true but cannot be used.
• chunkingMethod string, The method used to break the data into smaller chunks for transfer. The value can be set to sorted_by_identity_columns or if not set the chunking method will be chosen automatically.
• firstRowIsHeader boolean,
• exportAction string. The kind of export action you want to have the export execute. Set to "newsprsht" if you want a new worksheet inside a new spreadsheet. Set to "newwksht" if you want a new worksheet inside an existing spreadsheet. Set to "updatewksht" if you want to overwrite an existing worksheet inside an existing spreadsheet. Set to "appendwksht" if you want to append to the end of an existing worksheet inside an existing spreadsheet. Default is set to "newsprsht"

• sqlQuery string. If you are doing a Google Sheet export, this is your SQL query.

• contactLists string,

• soqlQuery string,

• includeDeletedRecords boolean,

Value

A list containing the following elements:

<table>
<thead>
<tr>
<th>id</th>
<th>integer,</th>
</tr>
</thead>
<tbody>
<tr>
<td>source</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• id integer, The ID of the table or file, if available.</td>
</tr>
<tr>
<td></td>
<td>• path string, The path of the dataset to sync from; for a database source, schema.tablename. If you are doing a Google Sheet export, this can be blank. This is a legacy parameter, it is recommended you use one of the following: databaseTable, file, googleWorksheet, salesforce</td>
</tr>
<tr>
<td></td>
<td>• databaseTable list . A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>-- schema string, The database schema name.</td>
</tr>
<tr>
<td></td>
<td>-- table string, The database table name.</td>
</tr>
<tr>
<td></td>
<td>-- useWithoutSchema boolean, This attribute is no longer available; defaults to false but cannot be used.</td>
</tr>
<tr>
<td></td>
<td>• file list . A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>-- id integer, The file id.</td>
</tr>
<tr>
<td></td>
<td>• googleWorksheet list . A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>-- spreadsheet string, The spreadsheet document name.</td>
</tr>
<tr>
<td></td>
<td>-- spreadsheetId string, The spreadsheet document id.</td>
</tr>
<tr>
<td></td>
<td>-- worksheet string, The worksheet tab name.</td>
</tr>
<tr>
<td></td>
<td>-- worksheetId integer, The worksheet tab id.</td>
</tr>
<tr>
<td></td>
<td>• salesforce list . A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>-- objectName string, The Salesforce object name.</td>
</tr>
<tr>
<td>destination</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• path string, The schema.tablename to sync to. If you are doing a Google Sheet export, this is the spreadsheet and sheet name separated by a period. i.e. if you have a spreadsheet named &quot;MySpreadsheet&quot; and a sheet called &quot;Sheet1&quot; this field would be &quot;MySpreadsheet.Sheet1&quot;. This is a legacy parameter, it is recommended you use one of the following: databaseTable, googleWorksheet</td>
</tr>
</tbody>
</table>
• databaseTable list. A list containing the following elements:
  – schema string, The database schema name.
  – table string, The database table name.
  – useWithoutSchema boolean, This attribute is no longer available; defaults to false but cannot be used.
• googleWorksheet list. A list containing the following elements:
  – spreadsheet string, The spreadsheet document name.
  – spreadsheetId string, The spreadsheet document id.
  – worksheet string, The worksheet tab name.
  – worksheetId integer, The worksheet tab id.

advancedOptions list, A list containing the following elements:
• maxErrors integer,
• existingTableRows string,
• diststyle string,
• distkey string,
• sortkey1 string,
• sortkey2 string,
• columnDelimiter string,
• columnOverrides object, Hash used for overriding auto-detected names and types, with keys being the index of the column being overridden.
• escaped boolean, If true, escape quotes with a backslash; otherwise, escape quotes by double-quoting. Defaults to false.
• identityColumn string,
• rowChunkSize integer,
• wipeDestinationTable boolean,
• truncateLongLines boolean,
• invalidCharReplacement string,
• verifyTableRowCounts boolean,
• partitionColumnName string, This parameter is deprecated
• partitionSchemaName string, This parameter is deprecated
• partitionTableName string, This parameter is deprecated
• partitionTablePartitionColumnMinName string, This parameter is deprecated
• partitionTablePartitionColumnMaxName string, This parameter is deprecated
• lastModifiedColumn string,
• mysqlCatalogMatchesSchema boolean, This attribute is no longer available; defaults to true but cannot be used.
• chunkingMethod string, The method used to break the data into smaller chunks for transfer. The value can be set to sorted_by_identity_columns or if not set the chunking method will be chosen automatically.
• firstRowIsHeader boolean,
• exportAction string. The kind of export action you want to have the export execute. Set to "newsprsht" if you want a new worksheet inside a new spreadsheet. Set to "newwksht" if you want a new worksheet inside an existing spreadsheet. Set to "updatewksht" if you want to overwrite an existing worksheet inside an existing spreadsheet. Set to "appendwksht" if you want to append to the end of an existing worksheet inside an existing spreadsheet. Default is set to "newsprsht"

• sqlQuery string. If you are doing a Google Sheet export, this is your SQL query.

• contactLists string,

• soqlQuery string,

• includeDeletedRecords boolean,

---

**imports_put**

*Update an import*

**Description**

Update an import

**Usage**

```javascript
imports_put(
    id,
    name,
    sync_type,
    is_outbound,
    source = NULL,
    destination = NULL,
    schedule = NULL,
    notifications = NULL,
    parent_id = NULL,
    next_run_at = NULL,
    time_zone = NULL
)
```

**Arguments**

- **id**
  - integer required. The ID for the import.

- **name**
  - string required. The name of the import.

- **sync_type**
  - string required. The type of sync to perform; one of Dbsync, AutoImport, GdocImport, GdocExport, and Salesforce.

- **is_outbound**
  - boolean required.

- **source**
  - list optional. A list containing the following elements:
    - **remoteHostId** integer,
• credentialId integer,
• additionalCredentials array, Array that holds additional credentials used for specific imports. For salesforce imports, the first and only element is the client credential id. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.

**destination** list optional. A list containing the following elements:
• remoteHostId integer,
• credentialId integer,
• additionalCredentials array, Array that holds additional credentials used for specific imports. For salesforce imports, the first and only element is the client credential id. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.

**schedule** list optional. A list containing the following elements:
• scheduled boolean, If the item is scheduled.
• scheduledDays array, Day based on numeric value starting at 0 for Sunday.
• scheduledHours array, Hours of the day it is scheduled on.
• scheduledMinutes array, Minutes of the day it is scheduled on.
• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

**notifications** list optional. A list containing the following elements:
• urls array, URLs to receive a POST request at job completion
• successEmailSubject string, Custom subject line for success e-mail.
• successEmailBody string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

**parent_id** integer optional. Parent id to trigger this import from

**next_run_at** string optional. The time of the next scheduled run.

**time_zone** string optional. The time zone of this import.
Value

A list containing the following elements:

- **name** string, The name of the import.
- **syncType** string, The type of sync to perform; one of Dbsync, AutoImport, GdocImport, GdocExport, and Salesforce.
- **source** list, A list containing the following elements:
  - remoteHostId integer,
  - credentialId integer,
  - additionalCredentials array, Array that holds additional credentials used for specific imports. For Salesforce imports, the first and only element is the client credential id. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.
  - name string,
- **destination** list, A list containing the following elements:
  - remoteHostId integer,
  - credentialId integer,
  - additionalCredentials array, Array that holds additional credentials used for specific imports. For Salesforce imports, the first and only element is the client credential id. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.
  - name string,
- **schedule** list, A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.
- **notifications** list, A list containing the following elements:
  - urls array, URLs to receive a POST request at job completion
  - successEmailSubject string, Custom subject line for success e-mail.
  - successEmailBody string, Custom body text for success e-mail, written in Markdown.
  - successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  - successEmailFromName string, Name from which success emails are sent; defaults to "Civis,"
  - successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  - failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

parentId integer, Parent id to trigger this import from

id integer, The ID for the import.

isOutbound boolean,

jobType string, The job type of this import.

syncs array, An array containing the following fields:
  - id integer,
  - source object,
  - destination object,
  - advancedOptions object,

state string,

createdAt string,

updatedAt string,

lastRun list, A list containing the following elements:
  - id integer,
  - state string,
  - createdAt string, The time that the run was queued.
  - startedAt string, The time that the run started.
  - finishedAt string, The time that the run completed.
  - error string, The error message for this run, if present.

user list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.

runningAs list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.

timeZone string, The time zone of this import.

hidden boolean, The hidden status of the item.

archived string, The archival status of the requested item(s).
**imports_put_archive**

*Update the archive status of this object*

**Description**

Update the archive status of this object

**Usage**

`imports_put_archive(id, status)`

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of the object.</td>
</tr>
<tr>
<td>status</td>
<td>boolean required. The desired archived status of the object.</td>
</tr>
</tbody>
</table>

**Value**

A list containing the following elements:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>string, The name of the import.</td>
</tr>
<tr>
<td>syncType</td>
<td>string, The type of sync to perform; one of Dbsync, AutoImport, GdocImport, GdocExport, and Salesforce.</td>
</tr>
<tr>
<td>source</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• remoteHostId integer,</td>
</tr>
<tr>
<td></td>
<td>• credentialId integer,</td>
</tr>
<tr>
<td></td>
<td>• additionalCredentials array, Array that holds additional credentials used for specific imports. For salesforce imports, the first and only element is the client credential id. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.</td>
</tr>
<tr>
<td></td>
<td>• name string.</td>
</tr>
<tr>
<td>destination</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• remoteHostId integer,</td>
</tr>
<tr>
<td></td>
<td>• credentialId integer,</td>
</tr>
<tr>
<td></td>
<td>• additionalCredentials array, Array that holds additional credentials used for specific imports. For salesforce imports, the first and only element is the client credential id. For DB Syncs, the first element is an SSL private key credential id, and the second element is the corresponding public key credential id.</td>
</tr>
<tr>
<td></td>
<td>• name string.</td>
</tr>
<tr>
<td>schedule</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• scheduled boolean, If the item is scheduled.</td>
</tr>
<tr>
<td></td>
<td>• scheduledDays array, Day based on numeric value starting at 0 for Sunday.</td>
</tr>
</tbody>
</table>
• scheduledHours array, Hours of the day it is scheduled on.
• scheduledMinutes array, Minutes of the day it is scheduled on.
• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
• urls array, URLs to receive a POST request at job completion
• successEmailSubject string, Custom subject line for success e-mail.
• successEmailBody string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

parentId integer, Parent id to trigger this import from
id integer, The ID for the import.
isOutbound boolean,
jobType string, The job type of this import.
syncs array, An array containing the following fields:
• id integer,
• source object,
• destination object,
• advancedOptions object,
state string,
createdAt string,
updatedAt string,
lastRun list, A list containing the following elements:
• id integer,
• state string,
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.
• error string, The error message for this run, if present.
user list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.
timeZone string, The time zone of this import.
hidden boolean, The hidden status of the item.
archived string, The archival status of the requested item(s).

---

**Description**

Replace all attributes of this CSV Import

**Usage**

```python
imports_put_files_csv(
    id,
    source,
    destination,
    first_row_is_header,
    name = NULL,
    column_delimiter = NULL,
    escaped = NULL,
    compression = NULL,
    existing_table_rows = NULL,
    max_errors = NULL,
    table_columns = NULL,
    loosen_types = NULL,
    execution = NULL,
    redshift_destination_options = NULL
)
```
### Arguments

**id**
- integer required. The ID for the import.

**source**
- list required. A list containing the following elements:
  - `fileIds` array, The file ID(s) to import, if importing Civis file(s).
  - `storagePath` list. A list containing the following elements:
    - `storageHostId` integer, The ID of the source storage host.
    - `credentialId` integer, The ID of the credentials for the source storage host.
    - `filePaths` array, The file or directory path(s) within the bucket from which to import. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/" If specifying a directory path, the job will import every file found under that path. All files must have the same column layout and file format (e.g., compression, columnDelimiter, etc.).

**destination**
- list required. A list containing the following elements:
  - `schema` string, The destination schema name.
  - `table` string, The destination table name.
  - `remoteHostId` integer, The ID of the destination database host.
  - `credentialId` integer, The ID of the credentials for the destination database.
  - `primaryKeys` array, A list of column(s) which together uniquely identify a row in the destination table. These columns must not contain NULL values. If the import mode is "upsert", this field is required; see the Civis Helpdesk article on "Advanced CSV Imports via the Civis API" for more information.
  - `lastModifiedKeys` array, A list of the columns indicating a record has been updated. If the destination table does not exist, and the import mode is "upsert", this field is required.

**first_row_is_header**
- boolean required. A boolean value indicating whether or not the first row of the source file is a header row.

**name**
- string optional. The name of the import.

**column_delimiter**
- string optional. The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

**escaped**
- boolean optional. A boolean value indicating whether or not the source file has quotes escaped with a backslash. Defaults to false.

**compression**
- string optional. The type of compression of the source file. Valid arguments are "gzip" and "none". Defaults to "none".

**existing_table_rows**
- string optional. The behavior if a destination table with the requested name already exists. One of "fail", "truncate", "append", "drop", or "upsert". Defaults to "fail".

**max_errors**
- integer optional. The maximum number of rows with errors to ignore before failing. This option is not supported for Postgres databases.

**table_columns**
- array optional. An array containing the following fields:
• name string, The column name.
• sqlType string, The SQL type of the column.

loosen_types boolean optional. If true, SQL types with precisions/lengths will have these values increased to accommodate data growth in future loads. Type loosening only occurs on table creation. Defaults to false.

execution string optional. In upsert mode, controls the movement of data in upsert mode. If set to "delayed", the data will be moved after a brief delay. If set to "immediate", the data will be moved immediately. In non-upsert modes, controls the speed at which detailed column stats appear in the data catalogue. Defaults to "delayed", to accommodate concurrent upserts to the same table and speedier non-upsert imports.

redshift_destination_options list optional. A list containing the following elements:
• diststyle string, The diststyle to use for the table. One of "even", "all", or "key".
• distkey string, Distkey for this table in Redshift
• sortkeys array, Sortkeys for this table in Redshift. Please provide a maximum of two.

Value
A list containing the following elements:

id integer, The ID for the import.
name string, The name of the import.
source list, A list containing the following elements:
• fileIds array, The file ID(s) to import, if importing Civis file(s).
• storagePath list . A list containing the following elements:
  – storageHostId integer, The ID of the source storage host.
  – credentialId integer, The ID of the credentials for the source storage host.
  – filePaths array, The file or directory path(s) within the bucket from which to import. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/"If specifying a directory path, the job will import every file found under that path. All files must have the same column layout and file format (e.g., compression, columnDelimiter, etc.).

destination list, A list containing the following elements:
• schema string, The destination schema name.
• table string, The destination table name.
• remoteHostId integer, The ID of the destination database host.
• credentialId integer, The ID of the credentials for the destination database.
• primaryKeys array, A list of column(s) which together uniquely identify a row in the destination table. These columns must not contain NULL values. If the import mode is "upsert", this field is required; see the Civis Helpdesk article on "Advanced CSV Imports via the Civis API" for more information.
• lastModifiedKeys array, A list of the columns indicating a record has been updated. If the destination table does not exist, and the import mode is "upsert", this field is required.

definition:
  - firstRowIsHeader: boolean, A boolean value indicating whether or not the first row of the source file is a header row.
  - columnDelimiter: string, The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".
  - escaped: boolean, A boolean value indicating whether or not the source file has quotes escaped with a backslash. Defaults to false.
  - compression: string, The type of compression of the source file. Valid arguments are "gzip" and "none". Defaults to "none".
  - existingTableRows: string, The behavior if a destination table with the requested name already exists. One of "fail", "truncate", "append", "drop", or "upsert". Defaults to "fail".
  - maxErrors: integer, The maximum number of rows with errors to ignore before failing. This option is not supported for Postgres databases.
  - tableColumns: array, An array containing the following fields:
    • name string, The column name.
    • sqlType string, The SQL type of the column.
  - loosenTypes: boolean, If true, SQL types with precisions/lengths will have these values increased to accommodate data growth in future loads. Type loosening only occurs on table creation. Defaults to false.
  - execution: string, In upsert mode, controls the movement of data in upsert mode. If set to "delayed", the data will be moved after a brief delay. If set to "immediate", the data will be moved immediately. In non-upsert modes, controls the speed at which detailed column stats appear in the data catalogue. Defaults to "delayed", to accommodate concurrent upserts to the same table and speedier non-upsert imports.
  - redshiftDestinationOptions: list, A list containing the following elements:
    • diststyle string, The diststyle to use for the table. One of "even", "all", or "key".
    • distkey string, Distkey for this table in Redshift
    • sortkeys array, Sortkeys for this table in Redshift. Please provide a maximum of two.
  - hidden: boolean, The hidden status of the item.
imports_put_files_csv_archive

Update the archive status of this object

Description

Update the archive status of this object

Usage

imports_put_files_csv_archive(id, status)

Arguments

id integer required. The ID of the object.
status boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id integer, The ID for the import.
name string, The name of the import.
source list, A list containing the following elements:
  • fileIds array, The file ID(s) to import, if importing Civis file(s).
  • storagePath list. A list containing the following elements:
    – storageHostId integer, The ID of the source storage host.
    – credentialId integer, The ID of the credentials for the source storage host.
    – filePaths array, The file or directory path(s) within the bucket from which to import. E.g. the file_path for "s3://mybucket/files/all/" would be "/files/all/" If specifying a directory path, the job will import every file found under that path. All files must have the same column layout and file format (e.g., compression, columnDelimiter, etc.).

destination list, A list containing the following elements:
  • schema string, The destination schema name.
  • table string, The destination table name.
  • remoteHostId integer, The ID of the destination database host.
  • credentialId integer, The ID of the credentials for the destination database.
  • primaryKeys array, A list of column(s) which together uniquely identify a row in the destination table. These columns must not contain NULL values. If the import mode is "upsert", this field is required; see the Civis Helpdesk article on "Advanced CSV Imports via the Civis API" for more information.
• lastModifiedKeys array. A list of the columns indicating a record has been updated. If the destination table does not exist, and the import mode is "upsert", this field is required.

firstRowIsHeader

boolean, A boolean value indicating whether or not the first row of the source file is a header row.

columnDelimiter

string, The column delimiter for the file. Valid arguments are "comma", "tab", and "pipe". Defaults to "comma".

escaped

boolean, A boolean value indicating whether or not the source file has quotes escaped with a backslash. Defaults to false.

compression

string, The type of compression of the source file. Valid arguments are "gzip" and "none". Defaults to "none".

existingTableRows

string, The behavior if a destination table with the requested name already exists. One of "fail", "truncate", "append", "drop", or "upsert". Defaults to "fail".

maxErrors

integer, The maximum number of rows with errors to ignore before failing. This option is not supported for Postgres databases.

tableColumns

array, An array containing the following fields:

• name string, The column name.
• sqlType string, The SQL type of the column.

loosenTypes

boolean, If true, SQL types with precisions/lengths will have these values increased to accommodate data growth in future loads. Type loosening only occurs on table creation. Defaults to false.

execution

string, In upsert mode, controls the movement of data in upsert mode. If set to "delayed", the data will be moved after a brief delay. If set to "immediate", the data will be moved immediately. In non-upsert modes, controls the speed at which detailed column stats appear in the data catalogue. Defaults to "delayed", to accommodate concurrent upserts to the same table and speedier non-upsert imports.

redshiftDestinationOptions

list, A list containing the following elements:

• diststyle string, The diststyle to use for the table. One of "even", "all", or "key".
• distkey string, Distkey for this table in Redshift
• sortkeys array, Sortkeys for this table in Redshift. Please provide a maximum of two.

hidden

boolean, The hidden status of the item.
**imports_put_projects**  
*Add an Import to a project*

**Description**

Add an Import to a project

**Usage**

```python
imports_put_projects(id, project_id)
```

**Arguments**

- **id** integer required. The ID of the Import.
- **project_id** integer required. The ID of the project.

**Value**

An empty HTTP response

---

**imports_put_shares_groups**  
*Set the permissions groups has on this object*

**Description**

Set the permissions groups has on this object

**Usage**

```python
imports_put_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```
**Arguments**

- **id**
  - integer required. The ID of the resource that is shared.

- **group_ids**
  - array required. An array of one or more group IDs.

- **permission_level**
  - string required. Options are: "read", "write", or "manage".

- **share_email_body**
  - string optional. Custom body text for e-mail sent on a share.

- **send_shared_email**
  - boolean optional. Send email to the recipients of a share.

**Value**

A list containing the following elements:

- **readers**
  - list, A list containing the following elements:
    - users array,
    - groups array,

- **writers**
  - list, A list containing the following elements:
    - users array,
    - groups array,

- **owners**
  - list, A list containing the following elements:
    - users array,
    - groups array,

- **totalUserShares**
  - integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

- **totalGroupShares**
  - integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**Imports Put Shares Users**

*Set the permissions users have on this object*

**Description**

Set the permissions users have on this object

**Usage**

```python
import import_put_shares_users

id,
user_ids,
permission_level,
share_email_body = NULL,
send_shared_email = NULL
```
**Imports put syncs**

**Update a sync**

**Arguments**

- **id** integer required. The ID of the resource that is shared.
- **user_ids** array required. An array of one or more user IDs.
- **permission_level** string required. Options are: "read", "write", or "manage".
- **share_email_body** string optional. Custom body text for e-mail sent on a share.
- **send_shared_email** boolean optional. Send email to the recipients of a share.

**Value**

A list containing the following elements:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,
- **writers** list, A list containing the following elements:
  - users array,
  - groups array,
- **owners** list, A list containing the following elements:
  - users array,
  - groups array,
- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

**Description**

Update a sync

**Usage**

```python
imports_put_syncs(id, sync_id, source, destination, advanced_options = NULL)
```
Arguments

id integer required. The ID of the import to fetch.
sync_id integer required. The ID of the sync to fetch.
source list required. A list containing the following elements:
  • path string, The path of the dataset to sync from; for a database source, schema.tablename. If you are doing a Google Sheet export, this can be blank. This is a legacy parameter, it is recommended you use one of the following: databaseTable, file, googleWorksheet, salesforce
  • databaseTable list . A list containing the following elements:
    – schema string, The database schema name.
    – table string, The database table name.
    – useWithoutSchema boolean, This attribute is no longer available; defaults to false but cannot be used.
  • file list .
  • googleWorksheet list . A list containing the following elements:
    – spreadsheet string, The spreadsheet document name.
    – spreadsheetId string, The spreadsheet document id.
    – worksheet string, The worksheet tab name.
    – worksheetId integer, The worksheet tab id.
  • salesforce list . A list containing the following elements:
    – objectName string, The Salesforce object name.

destination list required. A list containing the following elements:
  • path string, The schema.tablename to sync to. If you are doing a Google Sheet export, this is the spreadsheet and sheet name separated by a period. i.e. if you have a spreadsheet named "MySpreadsheet" and a sheet called "Sheet1" this field would be "MySpreadsheet.Sheel1". This is a legacy parameter, it is recommended you use one of the following: databaseTable, googleWorksheet
  • databaseTable list . A list containing the following elements:
    – schema string, The database schema name.
    – table string, The database table name.
    – useWithoutSchema boolean, This attribute is no longer available; defaults to false but cannot be used.
  • googleWorksheet list . A list containing the following elements:
    – spreadsheet string, The spreadsheet document name.
    – spreadsheetId string, The spreadsheet document id.
    – worksheet string, The worksheet tab name.
    – worksheetId integer, The worksheet tab id.

advanced_options list optional. A list containing the following elements:
  • maxErrors integer,
  • existingTableRows string,
• diststyle string,
• distkey string,
• sortkey1 string,
• sortkey2 string,
• columnDelimiter string,
• columnOverrides object, Hash used for overriding auto-detected names and types, with keys being the index of the column being overridden.
• escaped boolean, If true, escape quotes with a backslash; otherwise, escape quotes by double-quoting. Defaults to false.
• identityColumn string,
• rowChunkSize integer,
• wipeDestinationTable boolean,
• truncateLongLines boolean,
• invalidCharReplacement string,
• verifyTableRowCounts boolean,
• partitionColumnName string, This parameter is deprecated
• partitionSchemaName string, This parameter is deprecated
• partitionTableName string, This parameter is deprecated
• partitionTablePartitionColumnMinName string, This parameter is deprecated
• partitionTablePartitionColumnMaxName string, This parameter is deprecated
• lastModifiedColumn string,
• mysqlCatalogMatchesSchema boolean, This attribute is no longer available; defaults to true but cannot be used.
• chunkingMethod string, The method used to break the data into smaller chunks for transfer. The value can be set to sorted_by_identity_columns or if not set the chunking method will be chosen automatically.
• firstRowIsHeader boolean,
• exportAction string, The kind of export action you want to have the export execute. Set to "newsprsht" if you want a new worksheet inside a new spreadsheet. Set to "newwksht" if you want a new worksheet inside an existing spreadsheet. Set to "updatewksht" if you want to overwrite an existing worksheet inside an existing spreadsheet. Set to "appendwksht" if you want to append to the end of an existing worksheet inside an existing spreadsheet. Default is set to "newsprsht"
• sqlQuery string, If you are doing a Google Sheet export, this is your SQL query.
• contactLists string,
• soqlQuery string,
• includeDeletedRecords boolean,

Value
A list containing the following elements:
id integer,

source list, A list containing the following elements:
  • id integer, The ID of the table or file, if available.
  • path string, The path of the dataset to sync from; for a database source, schema.tablename. If you are doing a Google Sheet export, this can be blank. This is a legacy parameter, it is recommended you use one of the following: databaseTable, file, googleWorksheet, salesforce
  • databaseTable list. A list containing the following elements:
    – schema string, The database schema name.
    – table string, The database table name.
    – useWithoutSchema boolean, This attribute is no longer available; defaults to false but cannot be used.
  • file list. A list containing the following elements:
    – id integer, The file id.
  • googleWorksheet list. A list containing the following elements:
    – spreadsheet string, The spreadsheet document name.
    – spreadsheetId string, The spreadsheet document id.
    – worksheet string, The worksheet tab name.
    – worksheetId integer, The worksheet tab id.
  • salesforce list. A list containing the following elements:
    – objectName string, The Salesforce object name.

destination list, A list containing the following elements:
  • path string, The schema.tablename to sync to. If you are doing a Google Sheet export, this is the spreadsheet and sheet name separated by a period. i.e. if you have a spreadsheet named "MySpreadsheet" and a sheet called "Sheet1" this field would be "MySpreadsheet.Sheet1". This is a legacy parameter, it is recommended you use one of the following: databaseTable, googleWorksheet
  • databaseTable list. A list containing the following elements:
    – schema string, The database schema name.
    – table string, The database table name.
    – useWithoutSchema boolean, This attribute is no longer available; defaults to false but cannot be used.
  • googleWorksheet list. A list containing the following elements:
    – spreadsheet string, The spreadsheet document name.
    – spreadsheetId string, The spreadsheet document id.
    – worksheet string, The worksheet tab name.
    – worksheetId integer, The worksheet tab id.

advancedOptions list, A list containing the following elements:
  • maxErrors integer,
  • existingTableRows string,
  • diststyle string,
imports_put_syncs

• distkey string,
• sortkey1 string,
• sortkey2 string,
• columnDelimiter string,
• columnOverrides object, Hash used for overriding auto-detected names and types, with keys being the index of the column being overridden.
• escaped boolean, If true, escape quotes with a backslash; otherwise, escape quotes by double-quoting. Defaults to false.
• identityColumn string,
• rowChunkSize integer,
• wipeDestinationTable boolean,
• truncateLongLines boolean,
• invalidCharReplacement string,
• verifyTableRowCounts boolean,
• partitionColumnName string, This parameter is deprecated
• partitionSchemaName string, This parameter is deprecated
• partitionTableName string, This parameter is deprecated
• partitionTablePartitionColumnMinName string, This parameter is deprecated
• partitionTablePartitionColumnMaxName string, This parameter is deprecated
• lastModifiedColumn string,
• mysqlCatalogMatchesSchema boolean, This attribute is no longer available; defaults to true but cannot be used.
• chunkingMethod string, The method used to break the data into smaller chunks for transfer. The value can be set to sorted_by_identity_columns or if not set the chunking method will be chosen automatically.
• firstRowIsHeader boolean,
• exportAction string, The kind of export action you want to have the export execute. Set to "newsprsht" if you want a new worksheet inside a new spreadsheet. Set to "newwksht" if you want a new worksheet inside an existing spreadsheet. Set to "updatewksht" if you want to overwrite an existing worksheet inside an existing spreadsheet. Set to "appendwksht" if you want to append to the end of an existing worksheet inside an existing spreadsheet. Default is set to "newsprsht"
• sqlQuery string, If you are doing a Google Sheet export, this is your SQL query.
• contactLists string,
• soqlQuery string,
• includeDeletedRecords boolean,
**imports_put_syncs_archive**

*Update the archive status of this sync*

**Description**

Update the archive status of this sync

**Usage**

```
imports_put_syncs_archive(id, sync_id, status = NULL)
```

**Arguments**

- `id` integer required. The ID of the import to fetch.
- `sync_id` integer required. The ID of the sync to fetch.
- `status` boolean optional. The desired archived status of the sync.

**Value**

A list containing the following elements:

- `id` integer,
- `source` list, A list containing the following elements:
  - `id` integer, The ID of the table or file, if available.
  - `path` string, The path of the dataset to sync from; for a database source, `schema.tablename`. If you are doing a Google Sheet export, this can be blank. This is a legacy parameter, it is recommended you use one of the following: `databaseTable`, `file`, `googleWorksheet`, `salesforce`
  - `databaseTable` list. A list containing the following elements:
    - `schema` string, The database schema name.
    - `table` string, The database table name.
    - `useWithoutSchema` boolean, This attribute is no longer available; defaults to false but cannot be used.
  - `file` list. A list containing the following elements:
    - `id` integer, The file id.
  - `googleWorksheet` list. A list containing the following elements:
    - `spreadsheet` string, The spreadsheet document name.
    - `spreadsheetId` string, The spreadsheet document id.
    - `worksheet` string, The worksheet tab name.
    - `worksheetId` integer, The worksheet tab id.
  - `salesforce` list. A list containing the following elements:
    - `objectName` string, The Salesforce object name.
- `destination` list, A list containing the following elements:
• path string, The schematablename to sync to. If you are doing a Google Sheet export, this is the spreadsheet and sheet name separated by a period. i.e. if you have a spreadsheet named "MySpreadsheet" and a sheet called "Sheet1" this field would be "MySpreadsheet.Sheet1". This is a legacy parameter, it is recommended you use one of the following: databaseTable, googleWorksheet
  • databaseTable list. A list containing the following elements:
    – schema string, The database schema name.
    – table string, The database table name.
    – useWithoutSchema boolean, This attribute is no longer available; defaults to false but cannot be used.
  • googleWorksheet list. A list containing the following elements:
    – spreadsheet string, The spreadsheet document name.
    – spreadsheetId string, The spreadsheet document id.
    – worksheet string, The worksheet tab name.
    – worksheetId integer, The worksheet tab id.
• advancedOptions list, A list containing the following elements:
  • maxErrors integer,
  • existingTableRows string,
  • diststyle string,
  • distkey string,
  • sortkey1 string,
  • sortkey2 string,
  • columnDelimiter string,
  • columnOverrides object, Hash used for overriding auto-detected names and types, with keys being the index of the column being overridden.
  • escaped boolean, If true, escape quotes with a backslash; otherwise, escape quotes by double-quoting. Defaults to false.
  • identityColumn string,
  • rowChunkSize integer,
  • wipeDestinationTable boolean,
  • truncateLongLines boolean,
  • invalidCharReplacement string,
  • verifyTableRowCounts boolean,
  • partitionColumnName string, This parameter is deprecated
  • partitionSchemaName string, This parameter is deprecated
  • partitionTableName string, This parameter is deprecated
  • partitionTablePartitionColumnMinName string, This parameter is deprecated
  • partitionTablePartitionColumnMaxName string, This parameter is deprecated
  • lastModifiedColumn string,
jobs_delete_projects

Remove a Job from a project

Description
Remove a Job from a project

Usage
jobs_delete_projects(id, project_id)

Arguments
id                   integer required. The ID of the Job.
project_id           integer required. The ID of the project.

Value
An empty HTTP response
jobs_delete_runs

**Description**
Cancel a run

**Usage**

```
jobs_delete_runs(id, run_id)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required.</td>
<td>The ID of the Job.</td>
</tr>
<tr>
<td>run_id</td>
<td>integer required.</td>
<td>The ID of the Run.</td>
</tr>
</tbody>
</table>

**Value**

An empty HTTP response

jobs_delete_shares_groups

**Description**
Revoke the permissions a group has on this object

**Usage**

```
jobs_delete_shares_groups(id, group_id)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required.</td>
<td>The ID of the resource that is shared.</td>
</tr>
<tr>
<td>group_id</td>
<td>integer required.</td>
<td>The ID of the group.</td>
</tr>
</tbody>
</table>

**Value**

An empty HTTP response
jobs_delete_shares_users

Revoke the permissions a user has on this object

Description
Revoke the permissions a user has on this object

Usage
jobs_delete_shares_users(id, user_id)

Arguments
id: integer required. The ID of the resource that is shared.
user_id: integer required. The ID of the user.

Value
An empty HTTP response

jobs_get
Show basic job info

Description
Show basic job info

Usage
jobs_get(id)

Arguments
id: integer required. The ID for this job.

Value
A list containing the following elements:
id: integer,
name: string,
type: string,
fromTemplateId: integer,
state: string, Whether the job is idle, queued, running, cancelled, or failed.
jobs_get_runs

```javascript
createdAt string,
updatedAt string,
runs array, An array containing the following fields:
  • id integer,
  • state string,
  • createdAt string, The time that the run was queued.
  • startedAt string, The time that the run started.
  • finishedAt string, The time that the run completed.
  • error string, The error message for this run, if present.
lastRun list, A list containing the following elements:
  • id integer,
  • state string,
  • createdAt string, The time that the run was queued.
  • startedAt string, The time that the run started.
  • finishedAt string, The time that the run completed.
  • error string, The error message for this run, if present.
hidden boolean, The hidden status of the item.
archived string, The archival status of the requested item(s).
successEmailSubject string,
successEmailBody string,
runningAsUser string,
runByUser string,
schedule list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number
    of times to run per hour.
```

---

**jobs_get_runs**  
*Check status of a job*

### Description

Check status of a job

### Usage

```javascript
jobs_get_runs(id, run_id)
```
Arguments

**id**
integer required. The ID of the Job.

**run_id**
integer required. The ID of the Run.

Value

A list containing the following elements:

**id**
integer,

**state**
string,

**createdAt**
string. The time that the run was queued.

**startedAt**
string. The time that the run started.

**finishedAt**
string. The time that the run completed.

**error**
string. The error message for this run, if present.

---

**jobs_list**

*List Jobs*

---

Description

List Jobs

Usage

```python
jobs_list(
    state = NULL,
    type = NULL,
    q = NULL,
    permission = NULL,
    scheduled = NULL,
    hidden = NULL,
    archived = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```

Arguments

**state**
string optional. The job’s state. One or more of queued, running, succeeded, failed, and cancelled. Specify multiple values as a comma-separated list (e.g., "A,B").

**type**
string optional. The job’s type. Specify multiple values as a comma-separated list (e.g., "A,B").
jobs_list

- **q**: string optional. Query string to search on the id, name, and job type.
- **permission**: string optional. A permissions string, one of "read", "write", or "manage". Lists only jobs for which the current user has that permission.
- **scheduled**: boolean optional. If the item is scheduled.
- **hidden**: boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
- **archived**: string optional. The archival status of the requested item(s).
- **limit**: integer optional. Number of results to return. Defaults to its maximum of 50.
- **page_num**: integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order**: string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at.
- **order_dir**: string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

### Value

An array containing the following fields:

- **id**: integer,
- **name**: string,
- **type**: string,
- **fromTemplateId**: integer,
- **state**: string, Whether the job is idle, queued, running, cancelled, or failed.
- **createdAt**: string,
- **updatedAt**: string,
- **lastRun**: list, A list containing the following elements:
  - **id**: integer,
  - **state**: string,
  - **createdAt**: string, The time that the run was queued.
  - **startedAt**: string, The time that the run started.
  - **finishedAt**: string, The time that the run completed.
  - **error**: string, The error message for this run, if present.
- **archived**: string, The archival status of the requested item(s).
- **schedule**: list, A list containing the following elements:
  - **scheduled**: boolean, If the item is scheduled.
  - **scheduledDays**: array, Day based on numeric value starting at 0 for Sunday.
  - **scheduledHours**: array, Hours of the day it is scheduled on.
  - **scheduledMinutes**: array, Minutes of the day it is scheduled on.
  - **scheduledRunsPerHour**: integer, Alternative to scheduled minutes, number of times to run per hour.
Description

Show nested tree of children that this job triggers

Usage

`jobs_list_children(id)`

Arguments

`id` integer required. The ID for this job.

Value

A list containing the following elements:

- `id` integer,
- `name` string,
- `type` string,
- `fromTemplateId` integer,
- `state` string,
- `createdAt` string,
- `updatedAt` string,
- `runs` array, An array containing the following fields:
  - `id` integer,
  - `state` string,
  - `createdAt` string, The time that the run was queued.
  - `startedAt` string, The time that the run started.
  - `finishedAt` string, The time that the run completed.
  - `error` string, The error message for this run, if present.
- `lastRun` list, A list containing the following elements:
  - `id` integer,
  - `state` string,
  - `createdAt` string, The time that the run was queued.
  - `startedAt` string, The time that the run started.
  - `finishedAt` string, The time that the run completed.
  - `error` string, The error message for this run, if present.
- `children` array,
jobs_list_parents

Show chain of parents as a list that this job triggers from

Description

Show chain of parents as a list that this job triggers from

Usage

jobs_list_parents(id)

Arguments

id integer required. The ID for this job.

Value

A list containing the following elements:

id integer,
name string,
type string,
fromTemplateId integer,
state string, Whether the job is idle, queued, running, cancelled, or failed.
createdAt string,
updatedAt string,
runchistory array, An array containing the following fields:
  • id integer,
  • state string,
  • createdAt string, The time that the run was queued.
  • startedAt string, The time that the run started.
  • finishedAt string, The time that the run completed.
  • error string, The error message for this run, if present.
lastRun string, The hidden status of the item.
archived string, The archival status of the requested item(s).
successEmailSubject string,
successEmailBody string,
runningAsUser string,
runByUser string,
schedule list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

jobs_list_projects

List the projects a Job belongs to

Description

List the projects a Job belongs to

Usage

jobs_list_projects(id, hidden = NULL)

Arguments

id integer required. The ID of the Job.
hidden boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id integer, The ID for this project.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user's name.
  • username string, This user's username.
  • initials string, This user's initials.
  • online boolean, Whether this user is online.
name string, The name of this project.
description string, A description of the project.
**Description**

List runs for the given job

**Usage**

```plaintext
e jobs_list_runs( id, limit = NULL, page_num = NULL, order = NULL, order_dir = NULL )
```

**Arguments**

- `id` integer required. The ID for this job.
- `limit` integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
- `page_num` integer optional. Page number of the results to return. Defaults to the first page, 1.
- `order` string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
- `order_dir` string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.
Value
An array containing the following fields:

- **id**: integer,
- **state**: string,
- **createdAt**: string, The time that the run was queued.
- **startedAt**: string, The time that the run started.
- **finishedAt**: string, The time that the run completed.
- **error**: string, The error message for this run, if present.

---

**jobs_list_runs_logs**  
*Get the logs for a run*

Description
Get the logs for a run

Usage

```
jobs_list_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

- **id**: integer required. The ID of the job.
- **run_id**: integer required. The ID of the run.
- **last_id**: integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted. Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
- **limit**: integer optional. The maximum number of log messages to return. Default of 10000.

Value
An array containing the following fields:

- **id**: integer, The ID of the log.
- **createdAt**: string, The time the log was created.
- **message**: string, The log message.
- **level**: string, The level of the log. One of unknown,fatal,error,warn,info,debug.
jobs_list_runs_outputs

List the outputs for a run

Description

List the outputs for a run

Usage

jobs_list_runs_outputs(
  id,
  run_id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)

Arguments

id
  integer required. The ID of the job.
run_id
  integer required. The ID of the run.
limit
  integer optional. Number of results to return. Defaults to its maximum of 50.
page_num
  integer optional. Page number of the results to return. Defaults to the first page, 1.
order
  string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
order_dir
  string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

objectType
  string. The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId
  integer, The ID of the output.
name
  string, The name of the output.
link
  string, The hypermedia link to the output.
value
  string,
Description

List users and groups permissioned on this object

Usage

jobs_list_shares(id)

Arguments

id integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,

- **writers** list, A list containing the following elements:
  - users array,
  - groups array,

- **owners** list, A list containing the following elements:
  - users array,
  - groups array,

- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.
Description

List the workflows a job belongs to

Usage

jobs_list_workflows(id, archived = NULL)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required.</td>
</tr>
<tr>
<td>archived</td>
<td>string optional. The archival status of the requested item(s).</td>
</tr>
</tbody>
</table>

Value

An array containing the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer, The ID for this workflow.</td>
</tr>
<tr>
<td>name</td>
<td>string, The name of this workflow.</td>
</tr>
<tr>
<td>description</td>
<td>string, A description of the workflow.</td>
</tr>
<tr>
<td>valid</td>
<td>boolean, The validity of the workflow definition.</td>
</tr>
<tr>
<td>fileId</td>
<td>string, The file id for the s3 file containing the workflow configuration.</td>
</tr>
<tr>
<td>user</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• id integer, The ID of this user.</td>
</tr>
<tr>
<td></td>
<td>• name string, This user’s name.</td>
</tr>
<tr>
<td></td>
<td>• username string, This user’s username.</td>
</tr>
<tr>
<td></td>
<td>• initials string, This user’s initials.</td>
</tr>
<tr>
<td></td>
<td>• online boolean, Whether this user is online.</td>
</tr>
<tr>
<td>state</td>
<td>string, The state of the workflow. State is &quot;running&quot; if any execution is running, otherwise reflects most recent execution state.</td>
</tr>
<tr>
<td>schedule</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• scheduled boolean, If the item is scheduled.</td>
</tr>
<tr>
<td></td>
<td>• scheduledDays array, Day based on numeric value starting at 0 for Sunday.</td>
</tr>
<tr>
<td></td>
<td>• scheduledHours array, Hours of the day it is scheduled on.</td>
</tr>
<tr>
<td></td>
<td>• scheduledMinutes array, Minutes of the day it is scheduled on.</td>
</tr>
<tr>
<td></td>
<td>• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.</td>
</tr>
<tr>
<td>allowConcurrentExecutions</td>
<td>boolean, Whether the workflow can execute when already running.</td>
</tr>
<tr>
<td>timeZone</td>
<td>string, The time zone of this workflow.</td>
</tr>
</tbody>
</table>
nextExecutionAt
    string, The time of the next scheduled execution.
archived
    string, The archival status of the requested item(s).
createdAt
updatedAt

jobs_post_runs Run a job

Description
Run a job

Usage
jobs_post_runs(id)

Arguments
id integer required. The ID for this job.

Value
A list containing the following elements:

id integer,
state string,
createdAt string, The time that the run was queued.
startedAt string, The time that the run started.
finishedAt string, The time that the run completed.
error string, The error message for this run, if present.
**jobs_post_trigger_email**

*Generate and retrieve trigger email address*

**Description**

Generate and retrieve trigger email address

**Usage**

`jobs_post_trigger_email(id)`

**Arguments**

- `id` integer required. The ID for this job.

**Value**

A list containing the following elements:

- `triggerEmail` string, Email address which may be used to trigger this job to run.

---

**jobs_put_archive**

*Update the archive status of this object*

**Description**

Update the archive status of this object

**Usage**

`jobs_put_archive(id, status)`

**Arguments**

- `id` integer required. The ID of the object.
- `status` boolean required. The desired archived status of the object.
Value

A list containing the following elements:

- **id** integer,
- **name** string,
- **type** string,
- **fromTemplateId** integer,
- **state** string, Whether the job is idle, queued, running, cancelled, or failed.
- **createdAt** string,
- **updatedAt** string,
- **runs** array, An array containing the following fields:
  - id integer,
  - state string,
  - createdAt string, The time that the run was queued.
  - startedAt string, The time that the run started.
  - finishedAt string, The time that the run completed.
  - error string, The error message for this run, if present.
- **lastRun** list, A list containing the following elements:
  - id integer,
  - state string,
  - createdAt string, The time that the run was queued.
  - startedAt string, The time that the run started.
  - finishedAt string, The time that the run completed.
  - error string, The error message for this run, if present.
- **hidden** boolean, The hidden status of the item.
- **archived** string, The archival status of the requested item(s).
- **successEmailSubject** string,
- **successEmailBody** string,
- **runningAsUser** string,
- **runByUser** string,
- **schedule** list, A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.
jobs_put_projects  Add a Job to a project

Description

Add a Job to a project

Usage

jobs_put_projects(id, project_id)

Arguments

id  integer required. The ID of the Job.

project_id  integer required. The ID of the project.

Value

An empty HTTP response

---

jobs_put_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

jobs_put_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
Arguments

- **id** integer required. The ID of the resource that is shared.
- **group_ids** array required. An array of one or more group IDs.
- **permission_level** string required. Options are: "read", "write", or "manage".
- **share_email_body** string optional. Custom body text for e-mail sent on a share.
- **send_shared_email** boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array.
- **writers** list, A list containing the following elements:
  - users array,
  - groups array.
- **owners** list, A list containing the following elements:
  - users array,
  - groups array.
- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**jobs_put_shares_users**  
*Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```python
jobs_put_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```
Arguments

id \hspace{1cm} integer required. The ID of the resource that is shared.
user_ids \hspace{1cm} array required. An array of one or more user IDs.
permission_level \hspace{1cm} string required. Options are: "read", "write", or "manage".
share_email_body \hspace{1cm} string optional. Custom body text for e-mail sent on a share.
send_shared_email \hspace{1cm} boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers \hspace{1cm} list, A list containing the following elements:
  • users array,
  • groups array,

writers \hspace{1cm} list, A list containing the following elements:
  • users array,
  • groups array,

owners \hspace{1cm} list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares \hspace{1cm} integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares \hspace{1cm} integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

json_values_delete_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

json_values_delete_shares_groups(id, group_id)
**json_values_get**

**Arguments**

- **id**: integer required. The ID of the resource that is shared.
- **group_id**: integer required. The ID of the group.

**Value**

An empty HTTP response

---

**json_values_delete_shares_users**

Revoke the permissions a user has on this object

**Description**

Revoke the permissions a user has on this object

**Usage**

`json_values_delete_shares_users(id, user_id)`

**Arguments**

- **id**: integer required. The ID of the resource that is shared.
- **user_id**: integer required. The ID of the user.

**Value**

An empty HTTP response

---

**json_values_get**

Get details about a JSON Value

**Description**

Get details about a JSON Value

**Usage**

`json_values_get(id)`

**Arguments**

- **id**: integer required. The ID of the JSON Value.
**Value**

A list containing the following elements:

- `id` integer, The ID of the JSON Value.
- `name` string, The name of the JSON Value.
- `value` string, The deserialized JSON value.

---

**json_values_list_shares**

*List users and groups permissioned on this object*

---

**Description**

List users and groups permissioned on this object

**Usage**

`json_values_list_shares(id)`

**Arguments**

- `id` integer required. The ID of the resource that is shared.

**Value**

An array containing the following fields:

- `readers` list, A list containing the following elements:
  - users array,
  - groups array,
- `writers` list, A list containing the following elements:
  - users array,
  - groups array,
- `owners` list, A list containing the following elements:
  - users array,
  - groups array,
- `totalUserShares` integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- `totalGroupShares` integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.
json_values_patch  
Update some attributes of this JSON Value

Description
Update some attributes of this JSON Value

Usage
json_values_patch(id, name = NULL, value_str = NULL)

Arguments
- **id**  
  integer required. The ID of the JSON Value.
- **name**  
  string optional. The name of the JSON Value.
- **value_str**  
  string optional. The JSON value to store. Should be a serialized JSON string. Limited to 1000000 bytes.

Value
A list containing the following elements:
- **id**  
  integer, The ID of the JSON Value.
- **name**  
  string, The name of the JSON Value.
- **value**  
  string, The deserialized JSON value.

json_values_post  
Create a JSON Value

Description
Create a JSON Value

Usage
json_values_post(value_str, name = NULL)

Arguments
- **value_str**  
  string required. The JSON value to store. Should be a serialized JSON string. Limited to 1000000 bytes.
- **name**  
  string optional. The name of the JSON Value.
Value

A list containing the following elements:

- **id** integer, The ID of the JSON Value.
- **name** string, The name of the JSON Value.
- **value** string, The deserialized JSON value.

---

json_values_put_shares_groups

*Set the permissions groups has on this object*

Description

Set the permissions groups has on this object

Usage

```python
json_values_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
```

Arguments

- **id** integer required. The ID of the resource that is shared.
- **group_ids** array required. An array of one or more group IDs.
- **permission_level** string required. Options are: "read", "write", or "manage".
- **share_email_body** string optional. Custom body text for e-mail sent on a share.
- **send_shared_email** boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,

- **writers** list, A list containing the following elements:
  - users array,
json_values_put_shares_users

Set the permissions users have on this object

Description

Set the permissions users have on this object

Usage

```python
def json_values_put_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```

Arguments

- **id**: integer required. The ID of the resource that is shared.
- **user_ids**: array required. An array of one or more user IDs.
- **permission_level**: string required. Options are: "read", "write", or "manage".
- **share_email_body**: string optional. Custom body text for e-mail sent on a share.
- **send_shared_email**: boolean optional. Send email to the recipients of a share.

owners

list, A list containing the following elements:

- **users array**,
- **groups array**,

totalUserShares

integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares

integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.
Value
A list containing the following elements:

**readers**
list, A list containing the following elements:
  • users array,
  • groups array.

**writers**
list, A list containing the following elements:
  • users array,
  • groups array.

**owners**
list, A list containing the following elements:
  • users array,
  • groups array.

**totalUserShares**
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

**totalGroupShares**
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**match_targets_delete_shares_groups**

Revoke the permissions a group has on this object

---

Description
Revoke the permissions a group has on this object

Usage
match_targets_delete_shares_groups(id, group_id)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of the resource that is shared.</td>
</tr>
<tr>
<td>group_id</td>
<td>integer required. The ID of the group.</td>
</tr>
</tbody>
</table>

Value
An empty HTTP response
match_targets_delete_shares_users

Revoke the permissions a user has on this object

Description
Revoke the permissions a user has on this object

Usage
match_targets_delete_shares_users(id, user_id)

Arguments
id integer required. The ID of the resource that is shared.
user_id integer required. The ID of the user.

Value
An empty HTTP response

match_targets_get

Show Match Target info

Description
Show Match Target info

Usage
match_targets_get(id)

Arguments
id integer required. The ID of the match target

Value
A list containing the following elements:
id integer, The ID of the match target
name string, The name of the match target
targetFileName string, The name of the target file
createdAt string,
updatedAt string,
archived boolean, Whether the match target has been archived.
**match_targets_list**

*List match targets*

**Description**

List match targets

**Usage**

```plaintext
match_targets_list()
```

**Value**

An array containing the following fields:

- `id` integer, The ID of the match target
- `name` string, The name of the match target
- `targetFileName` string, The name of the target file
- `createdAt` string,
- `updatedAt` string,
- `archived` boolean, Whether the match target has been archived.

**match_targets_list_shares**

*List users and groups permissioned on this object*

**Description**

List users and groups permissioned on this object

**Usage**

```plaintext
match_targets_list_shares(id)
```

**Arguments**

- `id` integer required. The ID of the resource that is shared.
**Value**

An array containing the following fields:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,
- **writers** list, A list containing the following elements:
  - users array,
  - groups array,
- **owners** list, A list containing the following elements:
  - users array,
  - groups array,
- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

**match_targets_patch**

*Update a match target*

**Description**

Update a match target

**Usage**

`match_targets_patch(id, name = NULL, target_file_name = NULL, archived = NULL)`

**Arguments**

- **id** integer required. The ID of the match target
- **name** string optional. The name of the match target
- **target_file_name** string optional. The name of the target file
- **archived** boolean optional. Whether the match target has been archived.
Value

A list containing the following elements:

- id: integer, The ID of the match target
- name: string, The name of the match target
- targetFileName: string, The name of the target file
- createdAt: string,
- updatedAt: string,
- archived: boolean, Whether the match target has been archived.

match_targets_post  Create a new match target

Description

Create a new match target

Usage

match_targets_post(name, target_file_name = NULL, archived = NULL)

Arguments

- name: string required. The name of the match target
- target_file_name: string optional. The name of the target file
- archived: boolean optional. Whether the match target has been archived.

Value

A list containing the following elements:

- id: integer, The ID of the match target
- name: string, The name of the match target
- targetFileName: string, The name of the target file
- createdAt: string,
- updatedAt: string,
- archived: boolean, Whether the match target has been archived.
match_targets_put_archive

Update the archive status of this object

Description
Update the archive status of this object

Usage
match_targets_put_archive(id, status)

Arguments
id          integer required. The ID of the object.
status      boolean required. The desired archived status of the object.

Value
A list containing the following elements:
id          integer, The ID of the match target
name        string, The name of the match target
targetFileName string, The name of the target file
createdAt   string,
updatedAt   string,
archived    boolean, Whether the match target has been archived.

match_targets_put_shares_groups

Set the permissions groups has on this object

Description
Set the permissions groups has on this object

Usage
match_targets_put_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
match_targets_put_shares_users

Arguments

id  integer required. The ID of the resource that is shared.
group_ids  array required. An array of one or more group IDs.
permission_level  string required. Options are: "read", "write", or "manage".
share_email_body  string optional. Custom body text for e-mail sent on a share.
send_shared_email  boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers  list, A list containing the following elements:
  • users array,
  • groups array,
writers  list, A list containing the following elements:
  • users array,
  • groups array,
owners  list, A list containing the following elements:
  • users array,
  • groups array,
totalUserShares  integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares  integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

Description

Set the permissions users have on this object

Usage

match_targets_put_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
Arguments

id integer required. The ID of the resource that is shared.
user_ids array required. An array of one or more user IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers list, A list containing the following elements:
  • users array,
  • groups array,
writers list, A list containing the following elements:
  • users array,
  • groups array,
owners list, A list containing the following elements:
  • users array,
  • groups array,
totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

media_delete_optimizations_runs

Cancel a run

Description

Cancel a run

Usage

media_delete_optimizations_runs(id, run_id)
media_delete_optimizations_shares_groups

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td>ID of the optimization.</td>
</tr>
<tr>
<td>run_id</td>
<td>integer</td>
<td>ID of the run.</td>
</tr>
</tbody>
</table>

Value

An empty HTTP response

---

media_delete_optimizations_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

media_delete_optimizations_shares_groups(id, group_id)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td>ID of the resource that is shared.</td>
</tr>
<tr>
<td>group_id</td>
<td>integer</td>
<td>ID of the group.</td>
</tr>
</tbody>
</table>

Value

An empty HTTP response

---

media_delete_optimizations_shares_users

Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

media_delete_optimizations_shares_users(id, user_id)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td>ID of the resource that is shared.</td>
</tr>
<tr>
<td>user_id</td>
<td>integer</td>
<td>ID of the user.</td>
</tr>
</tbody>
</table>
media_delete_ratecards_shares_groups

Revoke the permissions a group has on this object

Description
Revoke the permissions a group has on this object

Usage
media_delete_ratecards_shares_groups(id, group_id)

Arguments
- id: integer required. The ID of the resource that is shared.
- group_id: integer required. The ID of the group.

Value
An empty HTTP response

media_delete_ratecards_shares_users

Revoke the permissions a user has on this object

Description
Revoke the permissions a user has on this object

Usage
media_delete_ratecards_shares_users(id, user_id)

Arguments
- id: integer required. The ID of the resource that is shared.
- user_id: integer required. The ID of the user.

Value
An empty HTTP response
media_delete_spot_orders_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

media_delete_spot_orders_shares_groups(id, group_id)

Arguments

id integer required. The ID of the resource that is shared.
group_id integer required. The ID of the group.

Value

An empty HTTP response

media_delete_spot_orders_shares_users

Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

media_delete_spot_orders_shares_users(id, user_id)

Arguments

id integer required. The ID of the resource that is shared.
user_id integer required. The ID of the user.

Value

An empty HTTP response
media_get_optimizations

*Show a single optimization*

**Description**

Show a single optimization

**Usage**

```
media_get_optimizations(id)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The optimization ID.</td>
</tr>
</tbody>
</table>

**Value**

A list containing the following elements:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer, The optimization ID.</td>
</tr>
<tr>
<td>author</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• id integer, The ID of this user.</td>
</tr>
<tr>
<td></td>
<td>• name string, This user’s name.</td>
</tr>
<tr>
<td></td>
<td>• username string, This user’s username.</td>
</tr>
<tr>
<td></td>
<td>• initials string, This user’s initials.</td>
</tr>
<tr>
<td></td>
<td>• online boolean, Whether this user is online.</td>
</tr>
<tr>
<td>name</td>
<td>string, The name of the optimization.</td>
</tr>
<tr>
<td>createdAt</td>
<td>string,</td>
</tr>
<tr>
<td>updatedAt</td>
<td>string,</td>
</tr>
<tr>
<td>finishedAt</td>
<td>string, The end time of the last run.</td>
</tr>
<tr>
<td>state</td>
<td>string, The state of the last run.</td>
</tr>
<tr>
<td>lastRunId</td>
<td>integer, The ID of the last run.</td>
</tr>
<tr>
<td>spotOrderId</td>
<td>integer, The ID for the spot order produced by the optimization.</td>
</tr>
<tr>
<td>archived</td>
<td>string, The archival status of the requested item(s).</td>
</tr>
<tr>
<td>reportLink</td>
<td>string, A link to the visual report for the optimization.</td>
</tr>
<tr>
<td>spotOrderLink</td>
<td>string, A link to the json version of the spot order.</td>
</tr>
<tr>
<td>fileLinks</td>
<td>array, Links to the csv and xml versions of the spot order.</td>
</tr>
<tr>
<td>runs</td>
<td>array, An array containing the following fields:</td>
</tr>
<tr>
<td></td>
<td>• marketId integer, The market ID.</td>
</tr>
<tr>
<td></td>
<td>• startDate string, The start date for the media run.</td>
</tr>
<tr>
<td></td>
<td>• endDate string, The end date for the media run.</td>
</tr>
</tbody>
</table>
media_get_optimizations_runs

Check status of a run

Description

Check status of a run

Usage

media_get_optimizations_runs(id, run_id)

Arguments

id   integer required. The ID of the optimization.
run_id   integer required. The ID of the run.
Value

A list containing the following elements:

- `id` (integer): The ID of the run.
- `optimizationId` (integer): The ID of the optimization.
- `state` (string): The state of the run, one of 'queued', 'running', 'succeeded', 'failed' or 'cancelled'.
- `isCancelRequested` (boolean): True if run cancel requested, else false.
- `startedAt` (string): The time the last run started at.
- `finishedAt` (string): The time the last run completed.
- `error` (string): The error, if any, returned by the run.

---

**media_get_ratecards**  
*Get a Ratecard*

---

Description

Get a Ratecard

Usage

`media_get_ratecards(id)`

Arguments

- `id` (integer): required.

Value

A list containing the following elements:

- `id` (integer): The ratecard ID.
- `filename` (string): Name of the ratecard file.
- `startOn` (string): First day to which the ratecard applies.
- `endOn` (string): Last day to which the ratecard applies.
- `dmaNumber` (integer): Number of the DMA associated with the ratecard.
- `archived` (string): The archival status of the requested item(s).
**media_get_spot_orders**  
*Show a single spot order*

**Description**  
Show a single spot order

**Usage**  
`media_get_spot_orders(id)`

**Arguments**

id  
integer required. The ID for the spot order.

**Value**

A list containing the following elements:

id  
integer, The ID for the spot order.

archived  
string, The archival status of the requested item(s).

csvS3Uri  
string, S3 URI for the spot order CSV file.

jsonS3Uri  
string, S3 URI for the spot order JSON file.

xmlArchiveS3Uri  
string, S3 URI for the spot order XML archive.

lastTransformJobId  
integer, ID of the spot order transformation job.

---

**media_list_dmas**  
*List all Designated Market Areas*

**Description**  
List all Designated Market Areas

**Usage**  
`media_list_dmas(name = NULL, number = NULL)`

**Arguments**

name  
string optional. If specified, will be used to filter the DMAs returned. Substring matching is supported with "%" and "*" wildcards (e.g., "name=%region%" will return both "region1" and "my region").

number  
integer optional. If specified, will be used to filter the DMAS by number.
**Value**

An array containing the following fields:

- **name** string, Name for the DMA region.
- **number** integer, Identifier number for a DMA.

---

**media_list_optimizations**

*List all optimizations*

**Description**

List all optimizations

**Usage**

```ruby
media_list_optimizations(
    archived = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```

**Arguments**

- **archived** string optional. The archival status of the requested item(s).
- **limit** integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
- **page_num** integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order** string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, author, name.
- **order_dir** string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

**Value**

An array containing the following fields:

- **id** integer, The optimization ID.
- **author** list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
media_list_optimizations_runs

- initials string, This user’s initials.
- online boolean, Whether this user is online.

name string, The name of the optimization.
createdAt string,
updatedAt string,
finishedAt string, The end time of the last run.
state string, The state of the last run.
lastRunId integer, The ID of the last run.
spotOrderId integer, The ID for the spot order produced by the optimization.
archived string, The archival status of the requested item(s).

---

media_list_optimizations_runs

List runs for the given optimization

---

**Description**

List runs for the given optimization

**Usage**

```python
media_list_optimizations_runs(
    id,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```

**Arguments**

- **id** integer required. The ID of the optimization.
- **limit** integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
- **page_num** integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order** string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
- **order_dir** string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.
Value

An array containing the following fields:

- **id**: integer, The ID of the run.
- **optimizationId**: integer, The ID of the optimization.
- **state**: string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
- **isCancelRequested**: boolean, True if run cancel requested, else false.
- **startedAt**: string, The time the last run started at.
- **finishedAt**: string, The time the last run completed.
- **error**: string, The error, if any, returned by the run.

---

media_list_optimizations_runs_logs

*Get the logs for a run*

Description

Get the logs for a run

Usage

```python
media_list_optimizations_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

Arguments

- **id**: integer required. The ID of the optimization.
- **run_id**: integer required. The ID of the run.
- **last_id**: integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted. Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
- **limit**: integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

- **id**: integer, The ID of the log.
- **createdAt**: string, The time the log was created.
- **message**: string, The log message.
- **level**: string, The level of the log. One of unknown,fatal,error,warn,info,debug.
media_list_optimizations_shares

List users and groups permissioned on this object

Description
List users and groups permissioned on this object

Usage
media_list_optimizations_shares(id)

Arguments
id integer required. The ID of the resource that is shared.

Value
An array containing the following fields:

readers list, A list containing the following elements:
• users array,
• groups array,

writers list, A list containing the following elements:
• users array,
• groups array,

owners list, A list containing the following elements:
• users array,
• groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.
media_list_ratecards  List all ratecards

Description
List all ratecards

Usage
media_list_ratecards(archived = NULL, filename = NULL, dma_number = NULL)

Arguments
- archived  string optional. The archival status of the requested item(s).
- filename  string optional. If specified, will be used to filter the ratecards returned. Sub-string matching is supported with "\%" and "\*" wildcards (e.g., "filename=\%ratecard\%" will return both "ratecard 1" and "my ratecard").
- dma_number  integer optional. If specified, will be used to filter the ratecards by DMA.

Value
An array containing the following fields:
- id  integer, The ratecard ID.
- filename  string, Name of the ratecard file.
- startDate  string, First day to which the ratecard applies.
- endDate  string, Last day to which the ratecard applies.
- dmaNumber  integer, Number of the DMA associated with the ratecard.
- archived  string, The archival status of the requested item(s).

media_list_ratecards_shares  List users and groups permissioned on this object

Description
List users and groups permissioned on this object

Usage
media_list_ratecards_shares(id)

Arguments
- id  integer required. The ID of the resource that is shared.
**Value**

An array containing the following fields:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,
- **writers** list, A list containing the following elements:
  - users array,
  - groups array,
- **owners** list, A list containing the following elements:
  - users array,
  - groups array,
- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**media_list_spot_orders**

*List all spot orders*

---

**Description**

List all spot orders

**Usage**

```plaintext
media_list_spot_orders(id = NULL, archived = NULL)
```

**Arguments**

- **id** integer optional. The ID for the spot order.
- **archived** string optional. The archival status of the requested item(s).

**Value**

An array containing the following fields:

- **id** integer, The ID for the spot order.
- **archived** string, The archival status of the requested item(s).
**media_list_spot_orders_shares**

*List users and groups permissioned on this object*

**Description**

List users and groups permissioned on this object

**Usage**

media_list_spot_orders_shares(id)

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of the resource that is shared.</td>
</tr>
</tbody>
</table>

**Value**

An array containing the following fields:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,

- **writers** list, A list containing the following elements:
  - users array,
  - groups array,

- **owners** list, A list containing the following elements:
  - users array,
  - groups array,

- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.
media_list_targets  

*List all Media Targets*

**Description**

List all Media Targets

**Usage**

```r
media_list_targets(name = NULL, identifier = NULL, data_source = NULL)
```

**Arguments**

- `name`  
  string optional. The name of the target.
- `identifier`  
  string optional. A unique identifier for this target.
- `data_source`  
  string optional. The source of viewership data for this target.

**Value**

An array containing the following fields:

- `name`  
  string, The name of the target.
- `identifier`  
  string, A unique identifier for this target.
- `dataSource`  
  string, The source of viewership data for this target.

media_patch_optimizations  

*Edit an existing optimization*

**Description**

Edit an existing optimization

**Usage**

```r
media_patch_optimizations(
  id,
  name = NULL,
  runs = NULL,
  programs = NULL,
  networks = NULL,
  exclude_programs = NULL,
  exclude_networks = NULL,
  time_slot_percentages = NULL
)
```
### Arguments

- **id**
  - integer required. The optimization ID.

- **name**
  - string optional. The name of the optimization.

- **runs**
  - array optional. An array containing the following fields:
    - marketId integer, The market ID.
    - startDate string, The start date for the media run.
    - endDate string, The end date for the media run.
    - forceCpm boolean, Whether to force optimization to use CPM data even if partition data is available.
    - reachAlpha number, A tuning parameter used to adjust RF.
    - syscodes array, The syscodes for the media run.
    - rateCards array, The ratecards for the media run.
    - constraints array, The constraints for the media run.

- **programs**
  - array optional. An array of programs that the Civis Media Optimizer either exclude or limit to. An error will be thrown if exclude_programs is not also set.

- **networks**
  - array optional. An array of networks that the Civis Media Optimizer either exclude or limit to. An error will be thrown if exclude_networks is not also set.

- **exclude_programs**
  - boolean optional. If Civis Media Optimizer should exclude the programs in the programs parameter. If this value is set to false, it will make the optimization limit itself to the programs supplied through the programs parameter. An error will be thrown if programs is not also set.

- **exclude_networks**
  - boolean optional. If Civis Media Optimizer should exclude the networks in the networks parameter. If this value is set to false, it will make the optimization limit itself to the networks supplied through the networks. An error will be thrown if networks is not also set.

- **time_slot_percentages**
  - list optional. The maximum amount of the budget spent on that particular day of the week, daypart, or specific time slot for broadcast and cable.

### Value

A list containing the following elements:

- **id**
  - integer, The optimization ID.

- **author**
  - list, A list containing the following elements:
    - id integer, The ID of this user.
    - name string, This user’s name.
    - username string, This user’s username.
    - initials string, This user’s initials.
    - online boolean, Whether this user is online.

- **name**
  - string, The name of the optimization.

- **createdAt**
  - string,
updatedAt string,
finishedAt string, The end time of the last run.
state string, The state of the last run.
lastRunId integer, The ID of the last run.
spotOrderId integer, The ID for the spot order produced by the optimization.
archived string, The archival status of the requested item(s).
reportLink string, A link to the visual report for the optimization.
spotOrderLink string, A link to the json version of the spot order.
fileLinks array, Links to the csv and xml versions of the spot order.
runs array, An array containing the following fields:
  • marketId integer, The market ID.
  • startDate string, The start date for the media run.
  • endDate string, The end date for the media run.
  • forceCpm boolean, Whether to force optimization to use CPM data even if
    partition data is available.
  • reachAlpha number, A tuning parameter used to adjust RF.
  • syscodes array, The syscodes for the media run.
  • rateCards array, The ratecards for the media run.
  • constraints array, The constraints for the media run.
programs array, An array of programs that the Civis Media Optimizer either exclude or
  limit to. An error will be thrown if exclude_programs is not also set.
networks array, An array of networks that the Civis Media Optimizer either exclude or
  limit to. An error will be thrown if exclude_networks is not also set.
excludePrograms boolean, If Civis Media Optimizer should exclude the programs in the programs
  parameter. If this value is set to false, it will make the optimization limit itself to
  the programs supplied through the programs parameter. An error will be thrown
  if programs is not also set.
excludeNetworks boolean, If Civis Media Optimizer should exclude the networks in the networks
  parameter. If this value is set to false, it will make the optimization limit itself to
  the networks supplied through the networks. An error will be thrown if networks
  is not also set.
timeSlotPercentages list, The maximum amount of the budget spent on that particular day of the
  week, daypart, or specific time slot for broadcast and cable.
Description

Update some attributes of this Ratecard

Usage

```r
media_patch_ratecards(
    id,
    filename = NULL,
    start_on = NULL,
    end_on = NULL,
    dma_number = NULL
)
```

Arguments

- **id**: integer required. The ratecard ID.
- **filename**: string optional. Name of the ratecard file.
- **start_on**: string optional. First day to which the ratecard applies.
- **end_on**: string optional. Last day to which the ratecard applies.
- **dma_number**: integer optional. Number of the DMA associated with the ratecard.

Value

A list containing the following elements:

- **id**: integer, The ratecard ID.
- **filename**: string, Name of the ratecard file.
- **startOn**: string, First day to which the ratecard applies.
- **endOn**: string, Last day to which the ratecard applies.
- **dmaNumber**: integer, Number of the DMA associated with the ratecard.
- **archived**: string, The archival status of the requested item(s).
media_post_optimizations

Create a new optimization

Description

Create a new optimization

Usage

media_post_optimizations(
   runs,
   name = NULL,
   programs = NULL,
   networks = NULL,
   exclude_programs = NULL,
   exclude_networks = NULL,
   time_slot_percentages = NULL
)

Arguments

* runs array required. An array containing the following fields:
  * marketId integer, The market ID.
  * startDate string, The start date for the media run.
  * endDate string, The end date for the media run.
  * forceCpm boolean, Whether to force optimization to use CPM data even if partition data is available.
  * reachAlpha number, A tuning parameter used to adjust RF.
  * syscodes array, The syscodes for the media run.
  * rateCards array, The ratecards for the media run.
  * constraints array, The constraints for the media run.

* name string optional. The name of the optimization.

* programs array optional. An array of programs that the Civis Media Optimizer either exclude or limit to. An error will be thrown if `exclude_programs` is not also set.

* networks array optional. An array of networks that the Civis Media Optimizer either exclude or limit to. An error will be thrown if `exclude_networks` is not also set.

* exclude_programs boolean optional. If Civis Media Optimizer should exclude the programs in the programs parameter. If this value is set to false, it will make the optimization limit itself to the programs supplied through the programs parameter. An error will be thrown if `programs` is not also set.
exclude_networks

boolean optional. If Civis Media Optimizer should exclude the networks in the networks parameter. If this value is set to false, it will make the optimization limit itself to the networks supplied through the networks. An error will be thrown if networks is not also set.

time_slot_percentages

list optional. The maximum amount of the budget spent on that particular day of the week, daypart, or specific time slot for broadcast and cable.

Value

A list containing the following elements:

id integer, The optimization ID.

author list, A list containing the following elements:

• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

name string, The name of the optimization.

createdAt string,

updatedAt string,

finishedAt string, The end time of the last run.

state string, The state of the last run.

lastRunId integer, The ID of the last run.

spotOrderId integer, The ID for the spot order produced by the optimization.

archived string, The archival status of the requested item(s).

reportLink string, A link to the visual report for the optimization.

spotOrderLink string, A link to the json version of the spot order.

fileLinks array, Links to the csv and xml versions of the spot order.

runs array, An array containing the following fields:

• marketId integer, The market ID.
• startDate string, The start date for the media run.
• endDate string, The end date for the media run.
• forceCpm boolean, Whether to force optimization to use CPM data even if partition data is available.
• reachAlpha number, A tuning parameter used to adjust RF.
• syscodes array, The syscodes for the media run.
• rateCards array, The ratecards for the media run.
• constraints array, The constraints for the media run.

programs array, An array of programs that the Civis Media Optimizer either exclude or limit to. An error will be thrown if exclude_programs is not also set.
media_post_optimizations_clone

Clone an existing optimization

Description
Clone an existing optimization

Usage
media_post_optimizations_clone(id)

Arguments
id integer required. The optimization ID.

Value
A list containing the following elements:

id integer, The optimization ID.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
name string, The name of the optimization.
createdAt string,
updatedAt string,
finishedAt string, The end time of the last run.
state string, The state of the last run.
lastRunId integer, The ID of the last run.
spotOrderId integer, The ID for the spot order produced by the optimization.
archived string, The archival status of the requested item(s).
reportLink string, A link to the visual report for the optimization.
spotOrderLink string, A link to the json version of the spot order.
fileLinks array, Links to the csv and xml versions of the spot order.
runs array, An array containing the following fields:
  • marketId integer, The market ID.
  • startDate string, The start date for the media run.
  • endDate string, The end date for the media run.
  • forceCpm boolean, Whether to force optimization to use CPM data even if partition data is available.
  • reachAlpha number, A tuning parameter used to adjust RF.
  • syscodes array, The syscodes for the media run.
  • rateCards array, The ratecards for the media run.
  • constraints array, The constraints for the media run.
programs array, An array of programs that the Civis Media Optimizer either exclude or limit to. An error will be thrown if exclude_programs is not also set.
networks array, An array of networks that the Civis Media Optimizer either exclude or limit to. An error will be thrown if exclude_networks is not also set.
excludePrograms boolean, If Civis Media Optimizer should exclude the programs in the programs parameter. If this value is set to false, it will make the optimization limit itself to the programs supplied through the programs parameter. An error will be thrown if programs is not also set.
excludeNetworks boolean, If Civis Media Optimizer should exclude the networks in the networks parameter. If this value is set to false, it will make the optimization limit itself to the networks supplied through the networks. An error will be thrown if networks is not also set.
timeSlotPercentages list, The maximum amount of the budget spent on that particular day of the week, daypart, or specific time slot for broadcast and cable.
media_post_optimizations_runs

Start a run

Description
Start a run

Usage
media_post_optimizations_runs(id)

Arguments
id integer required. The ID of the optimization.

Value
A list containing the following elements:

id integer, The ID of the run.
optimizationId integer, The ID of the optimization.
state string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested boolean, True if run cancel requested, else false.
startedAt string, The time the last run started at.
finishedAt string, The time the last run completed.
error string, The error, if any, returned by the run.

media_post_ratecards Create a Ratecard

Description
Create a Ratecard

Usage
media_post_ratecards(filename, start_on, end_on, dma_number)
Arguments

filename: string required. Name of the ratecard file.
start_on: string required. First day to which the ratecard applies.
end_on: string required. Last day to which the ratecard applies.
dma_number: integer required. Number of the DMA associated with the ratecard.

Value

A list containing the following elements:
id: integer, The ratecard ID.
filename: string, Name of the ratecard file.
startOn: string, First day to which the ratecard applies.
endOn: string, Last day to which the ratecard applies.
dmaNumber: integer, Number of the DMA associated with the ratecard.
archived: string, The archival status of the requested item(s).

media_post_spot_orders

Create a spot order

Description

Create a spot order

Usage

media_post_spot_orders(body = NULL)

Arguments

body: string optional. CSV body of a spot order.

Value

A list containing the following elements:
id: integer, The ID for the spot order.
archived: string, The archival status of the requested item(s).
csvS3Uri: string, S3 URI for the spot order CSV file.
jsonS3Uri: string, S3 URI for the spot order JSON file.
xmlArchiveS3Uri: string, S3 URI for the spot order XML archive.
lastTransformJobId: integer, ID of the spot order transformation job.
Description

Update the archive status of this object

Usage

media_put_optimizations_archive(id, status)

Arguments

id integer required. The ID of the object.
status boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id integer, The optimization ID.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
name string, The name of the optimization.
created_at string,
updatedAt string,
finishedAt string, The end time of the last run.
state string, The state of the last run.
lastRunId integer, The ID of the last run.
spotOrderId integer, The ID for the spot order produced by the optimization.
archived string, The archival status of the requested item(s).
reportLink string, A link to the visual report for the optimization.
spotOrderLink string, A link to the json version of the spot order.
fileLinks array, Links to the csv and xml versions of the spot order.
runs array, An array containing the following fields:
  • marketId integer, The market ID.
  • startDate string, The start date for the media run.
media_put_optimizations_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

media_put_optimizations_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
Arguments

- **id** integer required. The ID of the resource that is shared.
- **group_ids** array required. An array of one or more group IDs.
- **permission_level** string required. Options are: "read", "write", or "manage".
- **share_email_body** string optional. Custom body text for e-mail sent on a share.
- **send_shared_email** boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,
- **writers** list, A list containing the following elements:
  - users array,
  - groups array,
- **owners** list, A list containing the following elements:
  - users array,
  - groups array,
- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

Description

Set the permissions users have on this object

Usage

```python
media_put_optimizations_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```
media_put_ratecards

Arguments

id integer required. The ID of the resource that is shared.
user_ids array required. An array of one or more user IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers list, A list containing the following elements:
  • users array,
  • groups array,

writers list, A list containing the following elements:
  • users array,
  • groups array,

owners list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

Description

Replace all attributes of this Ratecard

Usage

media_put_ratecards(id, filename, start_on, end_on, dma_number)
Arguments

- **id**: integer required. The ratecard ID.
- **filename**: string required. Name of the ratecard file.
- **start_on**: string required. First day to which the ratecard applies.
- **end_on**: string required. Last day to which the ratecard applies.
- **dma_number**: integer required. Number of the DMA associated with the ratecard.

Value

A list containing the following elements:

- **id**: integer, The ratecard ID.
- **filename**: string, Name of the ratecard file.
- **startOn**: string, First day to which the ratecard applies.
- **endOn**: string, Last day to which the ratecard applies.
- **dmaNumber**: integer, Number of the DMA associated with the ratecard.
- **archived**: string, The archival status of the requested item(s).

---

**media_put_ratecards_archive**

*Update the archive status of this object*

---

Description

Update the archive status of this object

Usage

```
media_put_ratecards_archive(id, status)
```

Arguments

- **id**: integer required. The ID of the object.
- **status**: boolean required. The desired archived status of the object.

Value

A list containing the following elements:

- **id**: integer, The ratecard ID.
- **filename**: string, Name of the ratecard file.
- **startOn**: string, First day to which the ratecard applies.
- **endOn**: string, Last day to which the ratecard applies.
- **dmaNumber**: integer, Number of the DMA associated with the ratecard.
- **archived**: string, The archival status of the requested item(s).
media_put_ratecards_shares_groups

Set the permissions groups has on this object

Description
Set the permissions groups has on this object

Usage
media_put_ratecards_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)

Arguments
id integer required. The ID of the resource that is shared.
group_ids array required. An array of one or more group IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value
A list containing the following elements:
readers list, A list containing the following elements:
    • users array,
    • groups array,
writers list, A list containing the following elements:
    • users array,
    • groups array,
owners list, A list containing the following elements:
    • users array,
    • groups array,
totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares

integer. For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

media_put_ratecards_shares_users

Set the permissions users have on this object

Description

Set the permissions users have on this object

Usage

media_put_ratecards_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)

Arguments

id integer required. The ID of the resource that is shared.
user_ids array required. An array of one or more user IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers list, A list containing the following elements:
  • users array,
  • groups array,

writers list, A list containing the following elements:
  • users array,
  • groups array,

owners list, A list containing the following elements:
  • users array,
• groups array,

  totalUserShares
  integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

  totalGroupShares
  integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**media_put_spot_orders**  *Edit the specified spot order*

**Description**

Edit the specified spot order

**Usage**

media_put_spot_orders(id, body = NULL)

**Arguments**

- **id**
  - integer required. The ID for the spot order.

- **body**
  - string optional. CSV body of a spot order.

**Value**

A list containing the following elements:

- **id**
  - integer, The ID for the spot order.

- **archived**
  - string, The archival status of the requested item(s).

- **csvS3Uri**
  - string, S3 URI for the spot order CSV file.

- **jsonS3Uri**
  - string, S3 URI for the spot order JSON file.

- **xmlArchiveS3Uri**
  - string, S3 URI for the spot order XML archive.

- **lastTransformJobId**
  - integer, ID of the spot order transformation job.
media_put_spot_orders_archive

_Update the archive status of this object_

**Description**

Update the archive status of this object

**Usage**

```
media_put_spot_orders_archive(id, status)
```

**Arguments**

- **id**
  - integer required. The ID of the object.
- **status**
  - boolean required. The desired archived status of the object.

**Value**

A list containing the following elements:

- **id**
  - integer, The ID for the spot order.
- **archived**
  - string, The archival status of the requested item(s).
- **csvS3Uri**
  - string, S3 URI for the spot order CSV file.
- **jsonS3Uri**
  - string, S3 URI for the spot order JSON file.
- **xmlArchiveS3Uri**
  - string, S3 URI for the spot order XML archive.
- **lastTransformJobId**
  - integer, ID of the spot order transformation job.

media_put_spot_orders_shares_groups

_Set the permissions groups has on this object_

**Description**

Set the permissions groups has on this object

**Usage**

```
media_put_spot_orders_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```
media_put_spot_orders_shares_users

Arguments

id: integer required. The ID of the resource that is shared.
group_ids: array required. An array of one or more group IDs.
permission_level: string required. Options are: "read", "write", or "manage".
share_email_body: string optional. Custom body text for e-mail sent on a share.
send_shared_email: boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers: list, A list containing the following elements:
  • users array,
  • groups array,

writers: list, A list containing the following elements:
  • users array,
  • groups array,

owners: list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

media_put_spot_orders_shares_users

Set the permissions users have on this object

Description

Set the permissions users have on this object

Usage

media_put_spot_orders_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
**models_delete_builds**  

**Cancel a build**

**Description**

Cancel a build

**Usage**

```python
models_delete_builds(id, build_id)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td>integer required. The ID of the model.</td>
</tr>
<tr>
<td>build_id</td>
<td>integer</td>
<td>integer required. The ID of the build.</td>
</tr>
</tbody>
</table>

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td>integer required. The ID of the resource that is shared.</td>
</tr>
<tr>
<td>user_ids</td>
<td>array</td>
<td>An array of one or more user IDs.</td>
</tr>
<tr>
<td>permission_level</td>
<td>string</td>
<td>string required. Options are: &quot;read&quot;, &quot;write&quot;, or &quot;manage&quot;.</td>
</tr>
<tr>
<td>share_email_body</td>
<td>string</td>
<td>string optional. Custom body text for e-mail sent on a share.</td>
</tr>
<tr>
<td>send_shared_email</td>
<td>boolean</td>
<td>boolean optional. Send email to the recipients of a share.</td>
</tr>
</tbody>
</table>

**Value**

A list containing the following elements:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,
- **writers** list, A list containing the following elements:
  - users array,
  - groups array,
- **owners** list, A list containing the following elements:
  - users array,
  - groups array,
- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.
models_delete_shares_groups

Value

An empty HTTP response

models_delete_projects

Remove a Model from a project

Description

Remove a Model from a project

Usage

models_delete_projects(id, project_id)

Arguments

id integer required. The ID of the Model.
project_id integer required. The ID of the project.

Value

An empty HTTP response

models_delete_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

models_delete_shares_groups(id, group_id)

Arguments

id integer required. The ID of the resource that is shared.
group_id integer required. The ID of the group.

Value

An empty HTTP response
models_delete_shares_users

*Revoke the permissions a user has on this object*

**Description**

Revoke the permissions a user has on this object

**Usage**

`models_delete_shares_users(id, user_id)`

**Arguments**

- `id` integer required. The ID of the resource that is shared.
- `user_id` integer required. The ID of the user.

**Value**

An empty HTTP response

models_get

*Retrieve model configuration*

**Description**

Retrieve model configuration

**Usage**

`models_get(id)`

**Arguments**

- `id` integer required. The ID of the model.

**Value**

A list containing the following elements:

- `id` integer, The ID of the model.
- `tableName` string, The qualified name of the table containing the training set from which to build the model.
- `databaseId` integer, The ID of the database holding the training set table used to build the model.
credentialId integer, The ID of the credential used to read the target table. Defaults to the user’s default credential.

modelName string, The name of the model.

description string, A description of the model.

interactionTerms boolean, Whether to search for interaction terms.

boxCoxTransformation boolean, Whether to transform data so that it assumes a normal distribution. Valid only with continuous models.

modelTypeId integer, The ID of the model’s type.

primaryKey string, The unique ID (primary key) of the training dataset.

dependentVariable string, The dependent variable of the training dataset.

dependentVariableOrder array, The order of dependent variables, especially useful for Ordinal Modeling.

excludedColumns array, A list of columns which will be considered ineligible to be independent variables.

limitingSQL string, A custom SQL WHERE clause used to filter the rows used to build the model. (e.g., "id > 105").

activeBuildId integer, The ID of the current active build, the build used to score predictions.

crossValidationParameters list, Cross validation parameter grid for tree methods, e.g. "n_estimators": [100, 200, 500], "learning_rate": [0.01, 0.1], "max_depth": [2, 3].

numberOfFolds integer, Number of folds for cross validation. Default value is 5.

notifications list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

schedule list, A list containing the following elements:
• scheduled boolean, If the item is scheduled.
• scheduledDays array, Day based on numeric value starting at 0 for Sunday.
• scheduledHours array, Hours of the day it is scheduled on.
• scheduledMinutes array, Minutes of the day it is scheduled on.
• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

parentId integer, The ID of the parent job that will trigger this model.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

timeZone string, The time zone of this model.

lastRun list, A list containing the following elements:
• id integer,
• state string,
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.
• error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.

user list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

createdAt string, The time the model was created.

updatedAt string, The time the model was updated.

currentBuildState string, The status of the current model build. One of "succeeded", "failed", "queued", or "running,"or "idle", if no build has been attempted.

currentBuildException string, Exception message, if applicable, of the current model build.

builds array, An array containing the following fields:
• id integer, The ID of the model build.
• name string, The name of the model build.
• createdAt string, The time the model build was created.
• description string, A description of the model build.
• rootMeanSquaredError number, A key metric for continuous models. Nil for other model types.
• rSquaredError number, A key metric for continuous models. Nil for other model types.
• rocAuc number, A key metric for binary, multinomial, and ordinal models. Nil for other model types.

predictions array, An array containing the following fields:
• id integer, The ID of the model to which to apply the prediction.
• tableName string, The qualified name of the table on which to apply the predictive model.
• primaryKey array, The primary key or composite keys of the table being predicted.
• limitingSQL string, A SQL WHERE clause used to scope the rows to be predicted.
• outputTable string, The qualified name of the table to be created which will contain the model’s predictions.
• schedule object,
• state string, The status of the prediction. One of: "succeeded", "failed", "queued", or "running," or "idle", if no build has been attempted.

lastOutputLocation string, The output JSON for the last build.
archived string, The archival status of the requested item(s).

models_get_builds | Check status of a build

Description
Check status of a build

Usage
models_get_builds(id, build_id)

Arguments
id integer required. The ID of the model.
build_id integer required. The ID of the build.
models_list

Value

A list containing the following elements:

- **id**: integer, The ID of the model build.
- **state**: string, The state of the model build. one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
- **error**: string, The error, if any, returned by the build.
- **name**: string, The name of the model build.
- **createdAt**: string, The time the model build was created.
- **description**: string, A description of the model build.
- **rootMeanSquaredError**: number, A key metric for continuous models. Nil for other model types.
- **rSquaredError**: number, A key metric for continuous models. Nil for other model types.
- **rocAuc**: number, A key metric for binary, multinomial, and ordinal models. Nil for other model types.
- **transformationMetadata**: string, A string representing the full JSON output of the metadata for transformation of column names.
- **output**: string, A string representing the JSON output for the specified build. Only present when smaller than 10KB in size.
- **outputLocation**: string, A URL representing the location of the full JSON output for the specified build. The URL link will be valid for 5 minutes.

models_list

List

Description

List

Usage

```r
models_list(
  model_name = NULL,
  training_table_name = NULL,
  dependent_variable = NULL,
  author = NULL,
  status = NULL,
  hidden = NULL,
  archived = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>model_name</td>
<td>string optional</td>
<td>If specified, will be used to filter the models returned. Substring</td>
</tr>
<tr>
<td></td>
<td></td>
<td>matching is supported. (e.g., &quot;modelName=model&quot; will return both &quot;model1&quot;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>and &quot;my model&quot;).</td>
</tr>
<tr>
<td>training_table_name</td>
<td>string optional</td>
<td>If specified, will be used to filter the models returned by the training</td>
</tr>
<tr>
<td></td>
<td></td>
<td>dataset table name. Substring matching is supported. (e.g., &quot;trainingTableName=table&quot; will return both &quot;table1&quot; and &quot;my_table&quot;).</td>
</tr>
<tr>
<td>dependent_variable</td>
<td>string optional</td>
<td>If specified, will be used to filter the models returned by the dependent</td>
</tr>
<tr>
<td></td>
<td></td>
<td>variable column name. Substring matching is supported. (e.g., &quot;dependentVariable=predictor&quot; will return both &quot;predictor&quot; and &quot;my predictor&quot;).</td>
</tr>
<tr>
<td>author</td>
<td>string optional</td>
<td>If specified, return models from this author. It accepts a comma-separated</td>
</tr>
<tr>
<td></td>
<td></td>
<td>list of author ids.</td>
</tr>
<tr>
<td>status</td>
<td>string optional</td>
<td>If specified, returns models with one of these statuses. It accepts</td>
</tr>
<tr>
<td></td>
<td></td>
<td>a comma-separated list, possible values are 'running', 'failed', 'succeeded',</td>
</tr>
<tr>
<td></td>
<td></td>
<td>'idle', 'scheduled'.</td>
</tr>
<tr>
<td>hidden</td>
<td>boolean optional</td>
<td>If specified to be true, returns hidden items. Defaults to false,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>returning non-hidden items.</td>
</tr>
<tr>
<td>archived</td>
<td>string optional</td>
<td>The archival status of the requested item(s).</td>
</tr>
<tr>
<td>limit</td>
<td>integer optional</td>
<td>Number of results to return. Defaults to its maximum of 50.</td>
</tr>
<tr>
<td>page_num</td>
<td>integer optional</td>
<td>Page number of the results to return. Defaults to the first page, 1.</td>
</tr>
<tr>
<td>order</td>
<td>string optional</td>
<td>The field on which to order the result set. Defaults to updated_at.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Must be one of: updated_at, name, created_at, last_run.updated_at.</td>
</tr>
<tr>
<td>order_dir</td>
<td>string optional</td>
<td>Direction in which to sort, either asc (ascending) or desc (descending)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>defaulting to desc.</td>
</tr>
</tbody>
</table>

Value

An array containing the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td>The ID of the model.</td>
</tr>
<tr>
<td>tableName</td>
<td>string</td>
<td>The qualified name of the table containing the training set from which to</td>
</tr>
<tr>
<td></td>
<td></td>
<td>build the model.</td>
</tr>
<tr>
<td>databaseId</td>
<td>integer</td>
<td>The ID of the database holding the training set table used to build the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>model.</td>
</tr>
<tr>
<td>credentialId</td>
<td>integer</td>
<td>The ID of the credential used to read the target table. Defaults to the</td>
</tr>
<tr>
<td></td>
<td></td>
<td>user's default credential.</td>
</tr>
<tr>
<td>modelName</td>
<td>string</td>
<td>The name of the model.</td>
</tr>
<tr>
<td>description</td>
<td>string</td>
<td>A description of the model.</td>
</tr>
<tr>
<td>interactionTerms</td>
<td>boolean</td>
<td>Whether to search for interaction terms.</td>
</tr>
</tbody>
</table>
boxCoxTransformation
  boolean, Whether to transform data so that it assumes a normal distribution. Valid only with continuous models.

modelTypeId
  integer, The ID of the model’s type.

primaryKey
  string, The unique ID (primary key) of the training dataset.

dependentVariable
  string, The dependent variable of the training dataset.

dependentVariableOrder
  array, The order of dependent variables, especially useful for Ordinal Modeling.

excludedColumns
  array, A list of columns which will be considered ineligible to be independent variables.

limitingSQL
  string, A custom SQL WHERE clause used to filter the rows used to build the model. (e.g., "id > 105").

crossValidationParameters
  list, Cross validation parameter grid for tree methods, e.g. "n_estimators": [100, 200, 500], "learning_rate": [0.01, 0.1], "max_depth": [2, 3].

numberOfFolds
  integer, Number of folds for cross validation. Default value is 5.

schedule
  list, A list containing the following elements:
    • scheduled boolean, If the item is scheduled.
    • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
    • scheduledHours array, Hours of the day it is scheduled on.
    • scheduledMinutes array, Minutes of the day it is scheduled on.
    • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

parentId
  integer, The ID of the parent job that will trigger this model.

timeZone
  string, The time zone of this model.

lastRun
  list, A list containing the following elements:
    • id integer,
    • state string,
    • createdAt string, The time that the run was queued.
    • startedAt string, The time that the run started.
    • finishedAt string, The time that the run completed.
    • error string, The error message for this run, if present.

user
  list, A list containing the following elements:
    • id integer, The ID of this user.
    • name string, This user’s name.
    • username string, This user’s username.
    • initials string, This user’s initials.
    • online boolean, Whether this user is online.

createdAt
  string, The time the model was created.
models_list_builds

List builds for the given model

Description

List builds for the given model
Usage

models_list_builds(
    id,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)

Arguments

id  integer required. The ID of the model.
limit  integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num  integer optional. Page number of the results to return. Defaults to the first page, 1.
order  string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir  string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id  integer, The ID of the model build.
state  string, The state of the model build. One of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
error  string, The error, if any, returned by the build.
name  string, The name of the model build.
createdAt  string, The time the model build was created.
description  string, A description of the model build.
rootMeanSquaredError  number, A key metric for continuous models. Nil for other model types.
rSquaredError  number, A key metric for continuous models. Nil for other model types.
rocAuc  number, A key metric for binary, multinomial, and ordinal models. Nil for other model types.
transformationMetadata  string, A string representing the full JSON output of the metadata for transformation of column names
output  string, A string representing the JSON output for the specified build. Only present when smaller than 10KB in size.
outputLocation  string, A URL representing the location of the full JSON output for the specified build. The URL link will be valid for 5 minutes.
models_list_builds_logs

*Get the logs for a build*

**Description**

Get the logs for a build

**Usage**

```plaintext
models_list_builds_logs(id, build_id, last_id = NULL, limit = NULL)
```

**Arguments**

- `id` integer required. The ID of the model.
- `build_id` integer required. The ID of the build.
- `last_id` integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted. Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
- `limit` integer optional. The maximum number of log messages to return. Default of 10000.

**Value**

An array containing the following fields:

- `id` integer, The ID of the log.
- `createdAt` string, The time the log was created.
- `message` string, The log message.
- `level` string, The level of the log. One of unknown,fatal,error,warn,info,debug.

models_list_projects

*List the projects a Model belongs to*

**Description**

List the projects a Model belongs to

**Usage**

```plaintext
models_list_projects(id, hidden = NULL)
```
Arguments

id  integer required. The ID of the Model.

hidden  boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id  integer, The ID for this project.

author  list, A list containing the following elements:

• id  integer, The ID of this user.
• name  string, This user’s name.
• username  string, This user’s username.
• initials  string, This user’s initials.
• online  boolean, Whether this user is online.

name  string, The name of this project.

description  string, A description of the project.

users  array, An array containing the following fields:

• id  integer, The ID of this user.
• name  string, This user’s name.
• username  string, This user’s username.
• initials  string, This user’s initials.
• online  boolean, Whether this user is online.

autoShare  boolean,

createdAt  string,

updatedAt  string,

archived  string, The archival status of the requested item(s).

models_list_schedules  Show the model build schedule

Description

Show the model build schedule

Usage

models_list_schedules(id)

Arguments

id  integer required. The ID of the model associated with this schedule.
Value

A list containing the following elements:

- id integer, The ID of the model associated with this schedule.
- schedule list, A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

Description

List users and groups permissioned on this object

Usage

models_list_shares(id)

Arguments

- id integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

- readers list, A list containing the following elements:
  - users array,
  - groups array,
- writers list, A list containing the following elements:
  - users array,
  - groups array,
- owners list, A list containing the following elements:
  - users array,
  - groups array,
- totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.
**models_list_types**

List all available model types

**Description**

List all available model types

**Usage**

```python
models_list_types()
```

**Value**

An array containing the following fields:

- `id` integer, The ID of the model type.
- `algorithm` string, The name of the algorithm used to train the model.
- `dvType` string, The type of dependent variable predicted by the model.
- `fintAllowed` boolean, Whether this model type supports searching for interaction terms.

---

**models_put_archive**

Update the archive status of this object

**Description**

Update the archive status of this object

**Usage**

```python
models_put_archive(id, status)
```

**Arguments**

- `id` integer required. The ID of the object.
- `status` boolean required. The desired archived status of the object.

**Value**

A list containing the following elements:

- `id` integer, The ID of the model.
- `tableName` string, The qualified name of the table containing the training set from which to build the model.
- `databaseId` integer, The ID of the database holding the training set table used to build the model.
credentialId (integer): The ID of the credential used to read the target table. Defaults to the user’s default credential.

modelName (string): The name of the model.

description (string): A description of the model.

interactionTerms (boolean): Whether to search for interaction terms.

boxCoxTransformation (boolean): Whether to transform data so that it assumes a normal distribution. Valid only with continuous models.

modelTypeId (integer): The ID of the model’s type.

primaryKey (string): The unique ID (primary key) of the training dataset.

dependentVariable (string): The dependent variable of the training dataset.

dependentVariableOrder (array): The order of dependent variables, especially useful for Ordinal Modeling.

excludedColumns (array): A list of columns which will be considered ineligible to be independent variables.

limitingSQL (string): A custom SQL WHERE clause used to filter the rows used to build the model. (e.g., "id > 105").

activeBuildId (integer): The ID of the current active build, the build used to score predictions.

crossValidationParameters (list): Cross validation parameter grid for tree methods, e.g. "n_estimators": [100, 200, 500], "learning_rate": [0.01, 0.1], "max_depth": [2, 3].

numberOfFolds (integer): Number of folds for cross validation. Default value is 5.

notifications (list): A list containing the following elements:

- urls array: URLs to receive a POST request at job completion
- successEmailSubject string: Custom subject line for success e-mail.
- successEmailBody string: Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array: Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string: Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string: Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array: Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer: Stall warning emails will be sent after this amount of minutes.
- successOn boolean: If success email notifications are on.
- failureOn boolean: If failure email notifications are on.

schedule (list): A list containing the following elements:
• scheduled boolean, If the item is scheduled.
• scheduledDays array, Day based on numeric value starting at 0 for Sunday.
• scheduledHours array, Hours of the day it is scheduled on.
• scheduledMinutes array, Minutes of the day it is scheduled on.
• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

parentId integer, The ID of the parent job that will trigger this model.

runningAs list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

timeZone string, The time zone of this model.

lastRun list, A list containing the following elements:
  • id integer,
  • state string,
  • createdAt string, The time that the run was queued.
  • startedAt string, The time that the run started.
  • finishedAt string, The time that the run completed.
  • error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.

user list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

createdAt string, The time the model was created.

updatedAt string, The time the model was updated.

currentBuildState string, The status of the current model build. One of "succeeded", "failed", "queued", or "running," or "idle", if no build has been attempted.

currentBuildException string, Exception message, if applicable, of the current model build.

builds array, An array containing the following fields:
  • id integer, The ID of the model build.
  • name string, The name of the model build.
  • createdAt string, The time the model build was created.
  • description string, A description of the model build.
• rootMeanSquaredError number, A key metric for continuous models. Nil for other model types.
• rSquaredError number, A key metric for continuous models. Nil for other model types.
• rocAuc number, A key metric for binary, multinomial, and ordinal models. Nil for other model types.

predictions array, An array containing the following fields:
  • id integer, The ID of the model to which to apply the prediction.
  • tableName string, The qualified name of the table on which to apply the predictive model.
  • primaryKey array, The primary key or composite keys of the table being predicted.
  • limitingSQL string, A SQL WHERE clause used to scope the rows to be predicted.
  • outputTable string, The qualified name of the table to be created which will contain the model’s predictions.
  • schedule object,
  • state string, The status of the prediction. One of: "succeeded", "failed", "queued", or "running,"or "idle", if no build has been attempted.

lastOutputLocation string, The output JSON for the last build.
archived string, The archival status of the requested item(s).

models_put_projects Add a Model to a project

Description
Add a Model to a project

Usage
models_put_projects(id, project_id)

Arguments
id integer required. The ID of the Model.
project_id integer required. The ID of the project.

Value
An empty HTTP response
models_put_shares_groups

Set the permissions groups has on this object

Description
Set the permissions groups has on this object

Usage
models_put_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)

Arguments
id
    integer required. The ID of the resource that is shared.
group_ids
    array required. An array of one or more group IDs.
permission_level
    string required. Options are: "read", "write", or "manage".
share_email_body
    string optional. Custom body text for e-mail sent on a share.
send_shared_email
    boolean optional. Send email to the recipients of a share.

Value
A list containing the following elements:
readers
    list, A list containing the following elements:
    • users array,
    • groups array,
writers
    list, A list containing the following elements:
    • users array,
    • groups array,
owners
    list, A list containing the following elements:
    • users array,
    • groups array,
totalUserShares
    integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares
  integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

models_put_shares_users
  Set the permissions users have on this object

Description
  Set the permissions users have on this object

Usage
  models_put_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
  )

Arguments
  id
    integer required. The ID of the resource that is shared.
  user_ids
    array required. An array of one or more user IDs.
  permission_level
    string required. Options are: "read", "write", or "manage".
  share_email_body
    string optional. Custom body text for e-mail sent on a share.
  send_shared_email
    boolean optional. Send email to the recipients of a share.

Value
  A list containing the following elements:
  readers
    list, A list containing the following elements:
    • users array,
    • groups array,
  writers
    list, A list containing the following elements:
    • users array,
    • groups array,
  owners
    list, A list containing the following elements:
    • users array,
multipart_upload

- groups array,

  totalUserShares
  integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

  totalGroupShares
  integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

multipart_upload  Upload to files endpoint in parts.

Description

If a future::plan has been set, will be carried out in parallel.

Usage

multipart_upload(file, name = "", chunk_size = 32 * 1024, expires_at = NULL)

Arguments

  file  the file
  name  name of the upload, defaults to
  chunk_size  size of the chunks in bytes
  expires_at  when the file expires (default never).

notebooks_delete  Archive a Notebook (deprecated, use archiving endpoints instead)

Description

Archive a Notebook (deprecated, use archiving endpoints instead)

Usage

  notebooks_delete(id)

Arguments

  id  integer required.

Value

  An empty HTTP response
**notebooks_delete_deployments**

*Delete a Notebook deployment*

**Description**
Delete a Notebook deployment

**Usage**

```
notebooks_delete_deployments(notebook_id, deployment_id)
```

**Arguments**

- `notebook_id` integer required. The ID of the owning Notebook
- `deployment_id` integer required. The ID for this deployment

**Value**
An empty HTTP response

---

**notebooks_delete_projects**

*Remove a Notebook from a project*

**Description**
Remove a Notebook from a project

**Usage**

```
notebooks_delete_projects(id, project_id)
```

**Arguments**

- `id` integer required. The ID of the Notebook.
- `project_id` integer required. The ID of the project.

**Value**
An empty HTTP response
**notebooks_delete_shares_groups**

*Revoke the permissions a group has on this object*

**Description**

Revoke the permissions a group has on this object

**Usage**

```
notebooks_delete_shares_groups(id, group_id)
```

**Arguments**

- **id**
  - integer required. The ID of the resource that is shared.

- **group_id**
  - integer required. The ID of the group.

**Value**

An empty HTTP response

---

**notebooks_delete_shares_users**

*Revoke the permissions a user has on this object*

**Description**

Revoke the permissions a user has on this object

**Usage**

```
notebooks_delete_shares_users(id, user_id)
```

**Arguments**

- **id**
  - integer required. The ID of the resource that is shared.

- **user_id**
  - integer required. The ID of the user.

**Value**

An empty HTTP response
Description

Get a Notebook

Usage

notebooks_get(id)

Arguments

id integer required.

Value

A list containing the following elements:

id integer, The ID for this notebook.
name string, The name of this notebook.
language string, The kernel language of this notebook.
description string, The description of this notebook.
notebookUrl string, Time-limited URL to get the .ipynb file for this notebook.
notebookPreviewUrl string, Time-limited URL to get the .htm preview file for this notebook.
requirementsUrl string, Time-limited URL to get the requirements.txt file for this notebook.
fileId string, The file ID for the S3 file containing the .ipynb file.
requirementsFileId string, The file ID for the S3 file containing the requirements.txt file.
user list, A list containing the following elements:

• id integer, The ID of this user.
• name string, This user's name.
• username string, This user's username.
• initials string, This user's initials.
• online boolean, Whether this user is online.
dockerImageName string, The name of the docker image to pull from DockerHub.
dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
instanceType string, The EC2 instance type to deploy to.
memory integer, The amount of memory allocated to the notebook.
cpu integer, The amount of cpu allocated to the notebook.
notebooks_get

createdAt string,
updatedAt string,
mostRecentDeployment list, A list containing the following elements:
  • deploymentId integer, The ID for this deployment.
  • userId integer, The ID of the owner.
  • host string, Domain of the deployment.
  • name string, Name of the deployment.
  • dockerImageName string, The name of the docker image to pull from DockerHub.
  • dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
  • displayUrl string, A signed URL for viewing the deployed item.
  • instanceType string, The EC2 instance type requested for the deployment.
  • memory integer, The memory allocated to the deployment.
  • cpu integer, The cpu allocated to the deployment.
  • state string, The state of the deployment.
  • stateMessage string, A detailed description of the state.
  • createdAt string,
  • updatedAt string,
  • published boolean,
  • notebookId integer, The ID of owning Notebook

credentials array, A list of credential IDs to pass to the notebook.

environmentVariables list, Environment variables to be passed into the Notebook.

idleTimeout integer, How long the notebook will stay alive without any kernel activity.

gitRepoId integer, The ID of the git repository.

gitRepoUrl string, The url of the git repository.

gitRef string, The git reference if git repo is specified.

gitPath string, The path to the .ipynb file in the git repo that will be started up on notebook launch.

archived string, The archival status of the requested item(s).

hidden boolean, The hidden status of the item.
notebooks_get_deployments

*Get details about a Notebook deployment*

**Description**

Get details about a Notebook deployment

**Usage**

`notebooks_get_deployments(notebook_id, deployment_id)`

**Arguments**

- `notebook_id` integer required. The ID of the owning Notebook
- `deployment_id` integer required. The ID for this deployment

**Value**

A list containing the following elements:

- `deploymentId` integer, The ID for this deployment.
- `userId` integer, The ID of the owner.
- `host` string, Domain of the deployment.
- `name` string, Name of the deployment.
- `dockerImageName` string, The name of the docker image to pull from DockerHub.
- `dockerImageTag` string, The tag of the docker image to pull from DockerHub (default: latest).
- `displayUrl` string, A signed URL for viewing the deployed item.
- `instanceType` string, The EC2 instance type requested for the deployment.
- `memory` integer, The memory allocated to the deployment.
- `cpu` integer, The cpu allocated to the deployment.
- `state` string, The state of the deployment.
- `stateMessage` string, A detailed description of the state.
- `createdAt` string,
- `updatedAt` string,
- `published` boolean,
- `notebookId` integer, The ID of owning Notebook
notebooks_get_git_commits

Get file contents at commit_hash

Description

Get file contents at commit_hash

Usage

notebooks_get_git_commits(id, commit_hash)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of the file.</td>
</tr>
<tr>
<td>commit_hash</td>
<td>string required. The SHA (full or shortened) of the desired git commit.</td>
</tr>
</tbody>
</table>

Value

A list containing the following elements:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>content</td>
<td>string, The file’s contents.</td>
</tr>
<tr>
<td>type</td>
<td>string, The file’s type.</td>
</tr>
<tr>
<td>size</td>
<td>integer, The file’s size.</td>
</tr>
<tr>
<td>fileHash</td>
<td>string, The SHA of the file.</td>
</tr>
</tbody>
</table>

notebooks_list

List Notebooks

Description

List Notebooks

Usage

notebooks_list(
    hidden = NULL,
    archived = NULL,
    author = NULL,
    status = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
Arguments

- **hidden**: boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
- **archived**: string optional. The archival status of the requested item(s).
- **author**: string optional. If specified, return imports from this author. It accepts a comma-separated list of author IDs.
- **status**: string optional. If specified, returns notebooks with one of these statuses. It accepts a comma-separated list, possible values are 'running', 'pending', 'idle'.
- **limit**: integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
- **page_num**: integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order**: string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at.
- **order_dir**: string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

- **id**: integer, The ID for this notebook.
- **name**: string, The name of this notebook.
- **language**: string, The kernel language of this notebook.
- **description**: string, The description of this notebook.
- **user**: list, A list containing the following elements:
  - **id**: integer, The ID of this user.
  - **name**: string, This user’s name.
  - **username**: string, This user’s username.
  - **initials**: string, This user’s initials.
  - **online**: boolean, Whether this user is online.

- **createdAt**: string,
- **updatedAt**: string,
- **mostRecentDeployment**: list, A list containing the following elements:
  - **deploymentId**: integer, The ID for this deployment.
  - **userId**: integer, The ID of the owner.
  - **host**: string, Domain of the deployment.
  - **name**: string, Name of the deployment.
  - **dockerImageName**: string, The name of the docker image to pull from DockerHub.
  - **dockerImageTag**: string, The tag of the docker image to pull from DockerHub (default: latest).
• instanceType string, The EC2 instance type requested for the deployment.
• memory integer, The memory allocated to the deployment.
• cpu integer, The cpu allocated to the deployment.
• state string, The state of the deployment.
• stateMessage string, A detailed description of the state.
• createdAt string,
• updatedAt string,
• published boolean,
• notebookId integer, The ID of owning Notebook
archived string, The archival status of the requested item(s).

---

**notebooks_list_deployments**

*List deployments for a Notebook*

**Description**

List deployments for a Notebook

**Usage**

```python
notebooks_list_deployments(
    notebook_id,
    deployment_id = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```

**Arguments**

- **notebook_id** integer required. The ID of the owning Notebook
- **deployment_id** integer optional. The ID for this deployment
- **limit** integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
- **page_num** integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order** string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at.
- **order_dir** string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.
**Value**

An array containing the following fields:

- `deploymentId` integer, The ID for this deployment.
- `userId` integer, The ID of the owner.
- `host` string, Domain of the deployment.
- `name` string, Name of the deployment.
- `dockerImageName` string, The name of the docker image to pull from DockerHub.
- `dockerImageTag` string, The tag of the docker image to pull from DockerHub (default: latest).
- `instanceType` string, The EC2 instance type requested for the deployment.
- `memory` integer, The memory allocated to the deployment.
- `cpu` integer, The cpu allocated to the deployment.
- `state` string, The state of the deployment.
- `stateMessage` string, A detailed description of the state.
- `createdAt` string,
- `updatedAt` string,
- `published` boolean,
- `notebookId` integer, The ID of owning Notebook

---

**notebooks_list_deployments_logs**

*Get the logs for a Notebook deployment*

---

**Description**

Get the logs for a Notebook deployment

**Usage**

```python
notebooks_list_deployments_logs(
    id, deployment_id,
    start_at = NULL,
    end_at = NULL,
    limit = NULL
)
```
### notebooks_list_git

**Arguments**

- `id` (integer, required): The ID of the owning Notebook.
- `deployment_id` (integer, required): The ID for this deployment.
- `start_at` (string, optional): Log entries with a lower timestamp will be omitted.
- `end_at` (string, optional): Log entries with a higher timestamp will be omitted.
- `limit` (integer, optional): The maximum number of log messages to return. Default of 10000.

**Value**

An array containing the following fields:

- `message` (string): The log message.
- `stream` (string): The stream of the log. One of "stdout", "stderr".
- `createdAt` (string): The time the log was created.
- `source` (string): The source of the log. One of "system", "user".

### notebooks_list_git

*Get the git metadata attached to an item*

**Description**

Get the git metadata attached to an item.

**Usage**

`notebooks_list_git(id)`

**Arguments**

- `id` (integer, required): The ID of the file.

**Value**

A list containing the following elements:

- `gitRef` (string): A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
- `gitBranch` (string): The git branch that the file is on.
- `gitPath` (string): The path of the file in the repository.
- `gitRepo` (list): A list containing the following elements:
  - `id` (integer): The ID for this git repository.
  - `repoUrl` (string): The URL for this git repository.
  - `createdAt` (string):
  - `updatedAt` (string):
- `pullFromGit` (boolean): Automatically pull latest commit from git. Only works for scripts.
notebooks_list_git_commits

*Get the git commits for an item*

**Description**

Get the git commits for an item

**Usage**

`notebooks_list_git_commits(id)`

**Arguments**

- **id**: integer required. The ID of the file.

**Value**

A list containing the following elements:

- `commitHash`: string, The SHA of the commit.
- `authorName`: string, The name of the commit's author.
- `date`: string, The commit's timestamp.
- `message`: string, The commit message.

---

notebooks_list_projects

*List the projects a Notebook belongs to*

**Description**

List the projects a Notebook belongs to

**Usage**

`notebooks_list_projects(id, hidden = NULL)`

**Arguments**

- **id**: integer required. The ID of the Notebook.
- **hidden**: boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
Value

An array containing the following fields:

- **id** integer, The ID for this project.
- **author** list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.
- **name** string, The name of this project.
- **description** string, A description of the project.
- **users** array, An array containing the following fields:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.
- **autoShare** boolean.
- **createdAt** string,
- **updatedAt** string,
- **archived** string, The archival status of the requested item(s).

---

**notebooks_list_shares**  *List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

`notebooks_list_shares(id)`

Arguments

- **id** integer required. The ID of the resource that is shared.
Value

An array containing the following fields:

- **readers**: list, A list containing the following elements:
  - users array,
  - groups array,

- **writers**: list, A list containing the following elements:
  - users array,
  - groups array,

- **owners**: list, A list containing the following elements:
  - users array,
  - groups array,

- **totalUserShares**: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

- **totalGroupShares**: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**notebooks_list_update_links**

*Get URLs to update notebook*

---

**Description**

Get URLs to update notebook

**Usage**

```python
notebooks_list_update_links(id)
```

**Arguments**

- **id**: integer required.

**Value**

A list containing the following elements:

- **updateUrl**: string, Time-limited URL to PUT new contents of the .ipynb file for this notebook.
- **updatePreviewUrl**: string, Time-limited URL to PUT new contents of the .htm preview file for this notebook.
**Description**

Update some attributes of this Notebook

**Usage**

```r
notebooks_patch(
  id,
  name = NULL,
  language = NULL,
  description = NULL,
  file_id = NULL,
  requirements_file_id = NULL,
  requirements = NULL,
  docker_image_name = NULL,
  docker_image_tag = NULL,
  instance_type = NULL,
  memory = NULL,
  cpu = NULL,
  credentials = NULL,
  environment_variables = NULL,
  idle_timeout = NULL,
  git_repo_url = NULL,
  git_ref = NULL,
  git_path = NULL
)
```

**Arguments**

- **id** integer required. The ID for this notebook.
- **name** string optional. The name of this notebook.
- **language** string optional. The kernel language of this notebook.
- **description** string optional. The description of this notebook.
- **file_id** string optional. The file ID for the S3 file containing the .ipynb file.
- **requirements_file_id** string optional. The file ID for the S3 file containing the requirements.txt file.
- **requirements** string optional. The requirements txt file.
- **docker_image_name** string optional. The name of the docker image to pull from DockerHub.
- **docker_image_tag** string optional. The tag of the docker image to pull from DockerHub (default: latest).
instance_type: string optional. The EC2 instance type to deploy to.
memory: integer optional. The amount of memory allocated to the notebook.
cpu: integer optional. The amount of cpu allocated to the notebook.
credentials: array optional. A list of credential IDs to pass to the notebook.
environment_variables: list optional. Environment variables to be passed into the Notebook.
idle_timeout: integer optional. How long the notebook will stay alive without any kernel activity.
git_repo_url: string optional. The url of the git repository.
git_ref: string optional. The git reference if git repo is specified.
git_path: string optional. The path to the .ipynb file in the git repo that will be started up on notebook launch.

Value

A list containing the following elements:

- id: integer, The ID for this notebook.
- name: string, The name of this notebook.
- language: string, The kernel language of this notebook.
- description: string, The description of this notebook.
- notebookUrl: string, Time-limited URL to get the .ipynb file for this notebook.
- notebookPreviewUrl: string, Time-limited URL to get the .htm preview file for this notebook.
- requirementsUrl: string, Time-limited URL to get the requirements.txt file for this notebook.
- fileId: string, The file ID for the S3 file containing the .ipynb file.
- requirementsFileId: string, The file ID for the S3 file containing the requirements.txt file.
- user: list, A list containing the following elements:
  - id: integer, The ID of this user.
  - name: string, This user’s name.
  - username: string, This user’s username.
  - initials: string, This user’s initials.
  - online: boolean, Whether this user is online.
- dockerImageName: string, The name of the docker image to pull from DockerHub.
- dockerImageTag: string, The tag of the docker image to pull from DockerHub (default: latest).
- instanceType: string, The EC2 instance type to deploy to.
- memory: integer, The amount of memory allocated to the notebook.
- cpu: integer, The amount of cpu allocated to the notebook.
- createdAt: string,
updatedAt  string,
mostRecentDeployment list, A list containing the following elements:
  • deploymentId integer, The ID for this deployment.
  • userId integer, The ID of the owner.
  • host string, Domain of the deployment.
  • name string, Name of the deployment.
  • dockerImageName string, The name of the docker image to pull from DockerHub.
  • dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
  • displayUrl string, A signed URL for viewing the deployed item.
  • instanceType string, The EC2 instance type requested for the deployment.
  • memory integer, The memory allocated to the deployment.
  • cpu integer, The cpu allocated to the deployment.
  • state string, The state of the deployment.
  • stateMessage string, A detailed description of the state.
  • createdAt string,
  • updatedAt string,
  • published boolean,
  • notebookId integer, The ID of owning Notebook
credentials array, A list of credential IDs to pass to the notebook.
environmentVariables list, Environment variables to be passed into the Notebook.
idleTimeout integer, How long the notebook will stay alive without any kernel activity.
gitRepoId integer, The ID of the git repository.
gitRepoUrl string, The url of the git repository
gitRef string, The git reference if git repo is specified
gitPath string, The path to the .ipynb file in the git repo that will be started up on notebook launch
archived string, The archival status of the requested item(s).
hidden boolean, The hidden status of the item.

---

**notebooks_post**  

*Create a Notebook*

---

**Description**

Create a Notebook
Usage

notebooks_post(
    name = NULL,
    language = NULL,
    description = NULL,
    file_id = NULL,
    requirements_file_id = NULL,
    requirements = NULL,
    docker_image_name = NULL,
    docker_image_tag = NULL,
    instance_type = NULL,
    memory = NULL,
    cpu = NULL,
    credentials = NULL,
    environment_variables = NULL,
    idle_timeout = NULL,
    git_repo_url = NULL,
    git_ref = NULL,
    git_path = NULL,
    hidden = NULL
)

Arguments

name string optional. The name of this notebook.
language string optional. The kernel language of this notebook.
description string optional. The description of this notebook.
file_id string optional. The file ID for the S3 file containing the .ipynb file.
requirements_file_id string optional. The file ID for the S3 file containing the requirements.txt file.
requirements string optional. The requirements txt file.
docker_image_name string optional. The name of the docker image to pull from DockerHub.
docker_image_tag string optional. The tag of the docker image to pull from DockerHub (default: latest).
instance_type string optional. The EC2 instance type to deploy to.
memory integer optional. The amount of memory allocated to the notebook.
cpu integer optional. The amount of cpu allocated to the the notebook.
credentials array optional. A list of credential IDs to pass to the notebook.
environment_variables list optional. Environment variables to be passed into the Notebook.
idle_timeout integer optional. How long the notebook will stay alive without any kernel activity.
git_repo_url  string optional. The url of the git repository
git_ref  string optional. The git reference if git repo is specified
git_path  string optional. The path to the .ipynb file in the git repo that will be started up on notebook launch
hidden  boolean optional. The hidden status of the item.

Value
A list containing the following elements:

id  integer, The ID for this notebook.
name  string, The name of this notebook.
language  string, The kernel language of this notebook.
description  string, The description of this notebook.
notebookUrl  string, Time-limited URL to get the .ipynb file for this notebook.
notebookPreviewUrl  string, Time-limited URL to get the .htm preview file for this notebook.
requirementsUrl  string, Time-limited URL to get the requirements.txt file for this notebook.
fileId  string, The file ID for the S3 file containing the .ipynb file.
requirementsFileId  string, The file ID for the S3 file containing the requirements.txt file.
user  list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
dockerImageName  string, The name of the docker image to pull from DockerHub.
dockerImageTag  string, The tag of the docker image to pull from DockerHub (default: latest).
instanceType  string, The EC2 instance type to deploy to.
memory  integer, The amount of memory allocated to the notebook.
cpu  integer, The amount of cpu allocated to the the notebook.
createdAt  string,
updatedAt  string,
mostRecentDeployment  list, A list containing the following elements:
  • deploymentId integer, The ID for this deployment.
  • userId integer, The ID of the owner.
  • host string, Domain of the deployment.
  • name string, Name of the deployment.
• dockerImageName string, The name of the docker image to pull from DockerHub.
• dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
• displayUrl string, A signed URL for viewing the deployed item.
• instanceType string, The EC2 instance type requested for the deployment.
• memory integer, The memory allocated to the deployment.
• cpu integer, The cpu allocated to the deployment.
• state string, The state of the deployment.
• stateMessage string, A detailed description of the state.
• createdAt string,
• updatedAt string,
• published boolean,
• notebookId integer, The ID of owning Notebook

credentials array, A list of credential IDs to pass to the notebook.
environmentVariables list, Environment variables to be passed into the Notebook.
idleTimeout integer, How long the notebook will stay alive without any kernel activity.
gitRepoId integer, The ID of the git repository.
gitRepoUrl string, The url of the git repository.
gitRef string, The git reference if git repo is specified.
gitPath string, The path to the .ipynb file in the git repo that will be started up on notebook launch.
archived string, The archival status of the requested item(s).
hidden boolean, The hidden status of the item.

---

**Description**

Clone this Notebook

**Usage**

```cpp
tables_post_clone(id)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required.</td>
</tr>
</tbody>
</table>
Value

A list containing the following elements:

- `id` integer, The ID for this notebook.
- `name` string, The name of this notebook.
- `language` string, The kernel language of this notebook.
- `description` string, The description of this notebook.
- `notebookUrl` string, Time-limited URL to get the .ipynb file for this notebook.
- `notebookPreviewUrl` string, Time-limited URL to get the .htm preview file for this notebook.
- `requirementsUrl` string, Time-limited URL to get the requirements.txt file for this notebook.
- `fileId` string, The file ID for the S3 file containing the .ipynb file.
- `requirementsFileId` string, The file ID for the S3 file containing the requirements.txt file.
- `user` list, A list containing the following elements:
  - `id` integer, The ID of this user.
  - `name` string, This user’s name.
  - `username` string, This user’s username.
  - `initials` string, This user’s initials.
  - `online` boolean, Whether this user is online.
- `dockerImageName` string, The name of the docker image to pull from DockerHub.
- `dockerImageTag` string, The tag of the docker image to pull from DockerHub (default: latest).
- `instanceType` string, The EC2 instance type to deploy to.
- `memory` integer, The amount of memory allocated to the notebook.
- `cpu` integer, The amount of cpu allocated to the notebook.
- `createdAt` string,
- `updatedAt` string,
- `mostRecentDeployment` list, A list containing the following elements:
  - `deploymentId` integer, The ID for this deployment.
  - `userId` integer, The ID of the owner.
  - `host` string, Domain of the deployment.
  - `name` string, Name of the deployment.
  - `dockerImageName` string, The name of the docker image to pull from DockerHub.
  - `dockerImageTag` string, The tag of the docker image to pull from DockerHub (default: latest).
  - `displayUrl` string, A signed URL for viewing the deployed item.
  - `instanceType` string, The EC2 instance type requested for the deployment.
• memory integer, The memory allocated to the deployment.
• cpu integer, The cpu allocated to the deployment.
• state string, The state of the deployment.
• stateMessage string, A detailed description of the state.
• createdAt string,
• updatedAt string,
• published boolean,
• notebookId integer, The ID of owning Notebook

credentials array, A list of credential IDs to pass to the notebook.
environmentVariables list, Environment variables to be passed into the Notebook.
idleTimeout integer, How long the notebook will stay alive without any kernel activity.
giRepoId integer, The ID of the git repository.
gitRepoUrl string, The url of the git repository
gitRef string, The git reference if git repo is specified
gitPath string, The path to the .ipynb file in the git repo that will be started up on note-
book launch
archived string, The archival status of the requested item(s).
hidden boolean, The hidden status of the item.

---

notebooks_post_deployments

*Deploy a Notebook*

---

**Description**

Deploy a Notebook

**Usage**

```
notebooks_post_deployments(notebook_id, deployment_id = NULL, published = NULL)
```

**Arguments**

- **notebook_id** integer required. The ID of the owning Notebook
- **deployment_id** integer optional. The ID for this deployment
- **published** boolean optional.
Value

A list containing the following elements:

- deploymentId: integer, The ID for this deployment.
- userId: integer, The ID of the owner.
- host: string, Domain of the deployment.
- name: string, Name of the deployment.
- dockerImageName: string, The name of the docker image to pull from DockerHub.
- dockerImageTag: string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl: string, A signed URL for viewing the deployed item.
- instanceType: string, The EC2 instance type requested for the deployment.
- memory: integer, The memory allocated to the deployment.
- cpu: integer, The cpu allocated to the deployment.
- state: string, The state of the deployment.
- stateMessage: string, A detailed description of the state.
- createdAt: string,
- updatedAt: string,
- published: boolean,
- notebookId: integer, The ID of owning Notebook.

---

**notebooks_post_git_commits**

Commit and push a new version of the file

---

Description

Commit and push a new version of the file

Usage

```
notebooks_post_git_commits(id, content, message, file_hash)
```

Arguments

- **id**: integer required. The ID of the file.
- **content**: string required. The contents to commit to the file.
- **message**: string required. A commit message describing the changes being made.
- **file_hash**: string required. The full SHA of the file being replaced.
Value

A list containing the following elements:

- `content` (string): The file’s contents.
- `type` (string): The file’s type.
- `size` (integer): The file’s size.
- `fileHash` (string): The SHA of the file.

---

`notebooks_put`  
*Replace all attributes of this Notebook*

---

Description

Replace all attributes of this Notebook

Usage

```python
notebooks_put(
    id,  
    name = NULL,  
    language = NULL,  
    description = NULL,  
    file_id = NULL,  
    requirements_file_id = NULL,  
    requirements = NULL,  
    docker_image_name = NULL,  
    docker_image_tag = NULL,  
    instance_type = NULL,  
    memory = NULL,  
    cpu = NULL,  
    credentials = NULL,  
    environment_variables = NULL,  
    idle_timeout = NULL,  
    git_repo_url = NULL,  
    git_ref = NULL,  
    git_path = NULL
)
```

Arguments

- `id` (integer): Required. The ID for this notebook.
- `name` (string): Optional. The name of this notebook.
- `language` (string): Optional. The kernel language of this notebook.
- `description` (string): Optional. The description of this notebook.
- `file_id` (string): Optional. The file ID for the S3 file containing the .ipynb file.
requirements_file_id
string optional. The file ID for the S3 file containing the requirements.txt file.

requirements
string optional. The requirements.txt file.

docker_image_name
string optional. The name of the docker image to pull from DockerHub.

docker_image_tag
string optional. The tag of the docker image to pull from DockerHub (default: latest).

instance_type
string optional. The EC2 instance type to deploy to.

memory
integer optional. The amount of memory allocated to the notebook.

cpu
integer optional. The amount of cpu allocated to the notebook.

credentials
array optional. A list of credential IDs to pass to the notebook.

environment_variables
list optional. Environment variables to be passed into the Notebook.

idle_timeout
integer optional. How long the notebook will stay alive without any kernel activity.

git_repo_url
string optional. The url of the git repository

git_ref
string optional. The git reference if git repo is specified

git_path
string optional. The path to the .ipynb file in the git repo that will be started up on notebook launch

Value

A list containing the following elements:

id
integer, The ID for this notebook.

name
string, The name of this notebook.

language
string, The kernel language of this notebook.

description
string, The description of this notebook.

notebookUrl
string, Time-limited URL to get the .ipynb file for this notebook.

notebookPreviewUrl
string, Time-limited URL to get the .html preview file for this notebook.

requirementsUrl
string, Time-limited URL to get the requirements.txt file for this notebook.

fileId
string, The file ID for the S3 file containing the .ipynb file.

requirementsFileId
string, The file ID for the S3 file containing the requirements.txt file.

user
list, A list containing the following elements:

• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
- online boolean, Whether this user is online.

dockerImageName
string, The name of the docker image to pull from DockerHub.

dockerImageTag
string, The tag of the docker image to pull from DockerHub (default: latest).

instanceType
string, The EC2 instance type to deploy to.

memory
integer, The amount of memory allocated to the notebook.

cpu
integer, The amount of cpu allocated to the notebook.

createdAt
string,

updatedAt
string,

mostRecentDeployment
list, A list containing the following elements:
- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment.
- cpu integer, The cpu allocated to the deployment.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- createdAt string,
- updatedAt string,
- published boolean,
- notebookId integer, The ID of owning Notebook

credentials
array, A list of credential IDs to pass to the notebook.

environmentVariables
list, Environment variables to be passed into the Notebook.

idleTimeout
integer, How long the notebook will stay alive without any kernel activity.

gitRepoId
integer, The ID of the git repository.

gitRepoUrl
string, The url of the git repository

gitRef
string, The git reference if git repo is specified

gitPath
string, The path to the .ipynb file in the git repo that will be started up on notebook launch

archived
string, The archival status of the requested item(s).

hidden
boolean, The hidden status of the item.
notebooks_put_archive  
*Update the archive status of this object*

**Description**

Update the archive status of this object

**Usage**

`notebooks_put_archive(id, status)`

**Arguments**

- **id** integer required. The ID of the object.
- **status** boolean required. The desired archived status of the object.

**Value**

A list containing the following elements:

- **id** integer, The ID for this notebook.
- **name** string, The name of this notebook.
- **language** string, The kernel language of this notebook.
- **description** string, The description of this notebook.
- **notebookUrl** string, Time-limited URL to get the .ipynb file for this notebook.
- **notebookPreviewUrl** string, Time-limited URL to get the .htm preview file for this notebook.
- **requirementsUrl** string, Time-limited URL to get the requirements.txt file for this notebook.
- **fileId** string, The file ID for the S3 file containing the .ipynb file.
- **requirementsFileId** string, The file ID for the S3 file containing the requirements.txt file.
- **user** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **dockerImageName** string, The name of the docker image to pull from DockerHub.
- **dockerImageTag** string, The tag of the docker image to pull from DockerHub (default: latest).
- **instanceType** string, The EC2 instance type to deploy to.
- **memory** integer, The amount of memory allocated to the notebook.
cpu integer, The amount of cpu allocated to the notebook.

createdAt string,

updatedAt string,

mostRecentDeployment list, A list containing the following elements:
   • deploymentId integer, The ID for this deployment.
   • userId integer, The ID of the owner.
   • host string, Domain of the deployment.
   • name string, Name of the deployment.
   • dockerImageName string, The name of the docker image to pull from DockerHub.
   • dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
   • displayUrl string, A signed URL for viewing the deployed item.
   • instanceType string, The EC2 instance type requested for the deployment.
   • memory integer, The memory allocated to the deployment.
   • cpu integer, The cpu allocated to the deployment.
   • state string, The state of the deployment.
   • stateMessage string, A detailed description of the state.
   • createdAt string,
   • updatedAt string,
   • published boolean,
   • notebookId integer, The ID of owning Notebook

credentials array, A list of credential IDs to pass to the notebook.

environmentVariables list, Environment variables to be passed into the Notebook.

idleTimeout integer, How long the notebook will stay alive without any kernel activity.

gitRepoId integer, The ID of the git repository.

gitRepoUrl string, The url of the git repository

gitRef string, The git reference if git repo is specified

gitPath string, The path to the .ipynb file in the git repo that will be started up on notebook launch

archived string, The archival status of the requested item(s).

hidden boolean, The hidden status of the item.
Notebooks_put_git

Attach an item to a file in a git repo

Description

Attach an item to a file in a git repo

Usage

```r
notebooks_put_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  pull_from_git = NULL
)
``` 

Arguments

- **id** integer required. The ID of the file.
- **git_ref** string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
- **git_branch** string optional. The git branch that the file is on.
- **git_path** string optional. The path of the file in the repository.
- **git_repo_url** string optional. The URL of the git repository.
- **pull_from_git** boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

- **gitRef** string, A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
- **gitBranch** string, The git branch that the file is on.
- **gitPath** string, The path of the file in the repository.
- **gitRepo** list, A list containing the following elements:
  - **id** integer, The ID for this git repository.
  - **repoUrl** string, The URL for this git repository.
  - **createdAt** string,
  - **updatedAt** string,
- **pullFromGit** boolean, Automatically pull latest commit from git. Only works for scripts.
Add a Notebook to a project

Description

Add a Notebook to a project

Usage

```
notebooks_put_projects(id, project_id)
```

Arguments

- **id**: integer required. The ID of the Notebook.
- **project_id**: integer required. The ID of the project.

Value

An empty HTTP response

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

```
notebooks_put_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```
Arguments

- **id**: integer required. The ID of the resource that is shared.
- **group_ids**: array required. An array of one or more group IDs.
- **permission_level**: string required. Options are: "read", "write", or "manage".
- **share_email_body**: string optional. Custom body text for e-mail sent on a share.
- **send_shared_email**: boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- **readers**: list, A list containing the following elements:
  - users array,
  - groups array,
- **writers**: list, A list containing the following elements:
  - users array,
  - groups array,
- **owners**: list, A list containing the following elements:
  - users array,
  - groups array,
- **totalUserShares**: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- **totalGroupShares**: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**notebooks_put_shares_users**

*Set the permissions users have on this object*

---

**Description**

Set the permissions users have on this object

**Usage**

```python
notebooks_put_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```
Arguments

id
integer required. The ID of the resource that is shared.

user_ids
array required. An array of one or more user IDs.

permission_level
string required. Options are: "read", "write", or "manage".

share_email_body
string optional. Custom body text for e-mail sent on a share.

send_shared_email
boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers
list, A list containing the following elements:

- users array,
- groups array,

writers
list, A list containing the following elements:

- users array,
- groups array,

owners
list, A list containing the following elements:

- users array,
- groups array,

totalUserShares
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

notifications_list
Receive a stream of notifications as they come in

Description

Receive a stream of notifications as they come in

Usage

notifications_list(last_event_id = NULL, r = NULL, mock = NULL)
ontology_list

Arguments

last_event_id  string optional. allows browser to keep track of last event fired
r  string optional. specifies retry/reconnect timeout
mock  string optional. used for testing

Value

An empty HTTP response

ontology_list  List the ontology of column names Civis uses

Description

List the ontology of column names Civis uses

Usage

ontology_list(subset = NULL)

Arguments

subset  string optional. A subset of fields to return.

Value

An array containing the following fields:

key  string,
title  string,
desc  string, A description of this field.
aliases  array,
plot.civis_ml_classifier

Decile plot for classification with civis_ml

Description

Decile plot for classification with civis_ml

Usage

## S3 method for class 'civis_ml_classifier'
plot(x, name = NULL, ...)

Arguments

x                 civis_ml object.
name           Name of the class in a multiclass model to plot.
...               unused.

Value

A ggplot2 plot object is returned invisibly.

plot.civis_ml_regressor

Y-yhat plot for regression with civis_ml

Description

Y-yhat plot for regression with civis_ml

Usage

## S3 method for class 'civis_ml_regressor'
plot(x, ...)

Arguments

x                 civis_ml object
...               unused

Value

A ggplot2 plot object is returned invisibly.
predictions_get

Show the specified prediction

Description
Show the specified prediction

Usage
predictions_get(id)

Arguments
id
integer required. The ID of the prediction.

Value
A list containing the following elements:

- id integer, The ID of the prediction.
- modelId integer, The ID of the model used for this prediction.
- scoredTableId integer, The ID of the source table for this prediction.
- scoredTableName string, The name of the source table for this prediction.
- outputTableName string, The name of the output table for this prediction.
- state string, The state of the last run of this prediction.
- error string, The error, if any, of the last run of this prediction.
- startedAt string, The start time of the last run of this prediction.
- finishedAt string, The end time of the last run of this prediction.
- lastRun string, A list containing the following elements:
  - id integer,
  - createdAt string, The time that the run was queued.
  - startedAt string, The time that the run started.
  - finishedAt string, The time that the run completed.
  - error string, The error message for this run, if present.
- scoredTables array, An array containing the following fields:
  - id integer, The ID of the table with created predictions.
  - schema string, The schema of table with created predictions.
  - name string, The name of table with created predictions.
  - createdAt string, The time when the table with created predictions was created.
predictions_list

- scoreStats array, An array of metrics on the created predictions.
- schedule list, A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.
- limitingSQL string, A SQL WHERE clause used to scope the rows to be predicted.
- primaryKey array, The primary key or composite keys of the table being predicted.

predictions_list List predictions

Description
List predictions

Usage
predictions_list(model_id = NULL)

Arguments
  model_id integer optional. If specified, only return predictions associated with this model ID.

Value
An array containing the following fields:
  id integer, The ID of the prediction.
  modelId integer, The ID of the model used for this prediction.
  scoredTableId integer, The ID of the source table for this prediction.
  scoredTableName string, The name of the source table for this prediction.
  outputTableName string, The name of the output table for this prediction.
  state string, The state of the last run of this prediction.
  error string, The error, if any, of the last run of this prediction.
  startedAt string, The start time of the last run of this prediction.
  finishedAt string, The end time of the last run of this prediction.
  lastRun list, A list containing the following elements:
predictions_list_schedules

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

predictions_list_schedules

Show the prediction schedule

Description
Show the prediction schedule

Usage
predictions_list_schedules(id)

Arguments
id integer required. ID of the prediction associated with this schedule.

Value
A list containing the following elements:

id integer, ID of the prediction associated with this schedule.
schedule list, A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.
scoreOnModelBuild boolean, Whether the prediction will run after a rebuild of the associated model.
print.civis_api  
*Print results from a Civis API call*

**Description**

Print results from a Civis API call

**Usage**

```r
## S3 method for class 'civis_api'
print(x, ...)
```

**Arguments**

- **x**  
  A `civis_api` response.
- **...**  
  Further arguments passed to `str`

**Value**

The `civis_api` object, invisibly.

**Examples**

```r
## Not run:
query_civis("SELECT * FROM schema.my_table", "database", preview_rows = 100)
csv_to_civis("file.csv", "my_database", "schema.my_table")
print(databases_list(), max = 5)

## End(Not run)
```

---

projects_delete  
*Archive a project (deprecated, use the /archive endpoint instead)*

**Description**

Archive a project (deprecated, use the /archive endpoint instead)

**Usage**

```r
projects_delete(project_id)
```

**Arguments**

- **project_id**  
  integer required.

**Value**

An empty HTTP response
projects_delete_parent_projects

**Description**
Remove an item from a Parent Project

**Usage**

```python
projects_delete_parent_projects(id, parent_project_id)
```

**Arguments**

- `id` integer required. The ID of the item.
- `parent_project_id` integer required. The ID of the Parent Project.

**Value**

An empty HTTP response

projects_delete_shares_groups

**Description**
Revoke the permissions a group has on this object

**Usage**

```python
projects_delete_shares_groups(id, group_id)
```

**Arguments**

- `id` integer required. The ID of the resource that is shared.
- `group_id` integer required. The ID of the group.

**Value**

An empty HTTP response
projects_delete_shares_users

Revoke the permissions a user has on this object

Description
Revoke the permissions a user has on this object

Usage
projects_delete_shares_users(id, user_id)

Arguments
- id integer required. The ID of the resource that is shared.
- user_id integer required. The ID of the user.

Value
An empty HTTP response

projects_get
Get a detailed view of a project and the objects in it

Description
Get a detailed view of a project and the objects in it

Usage
projects_get(project_id)

Arguments
- project_id integer required.

Value
A list containing the following elements:
- id integer, The ID for this project.
- author list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

name string, The name of this project.

description string, A description of the project.

users array, An array containing the following fields:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

autoShare boolean,
createdAt string,
updatedAt string,

tables array, An array containing the following fields:
• schema string,
• name string,
• rowCount integer,
• columnCount integer,
• createdAt string,
• updatedAt string,

surveys array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,

scripts array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• name string,
• type string,
• finishedAt string,
• state string,
• lastRun object,

imports array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• name string,
• type string,
• finishedAt string,
- state string,
- lastRun object,

exports array, An array containing the following fields:
- id integer, The item’s ID.
- createdAt string,
- updatedAt string,
- name string,
- type string,
- finishedAt string,
- state string,
- lastRun object,

models array, An array containing the following fields:
- id integer, The item’s ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,

notebooks array, An array containing the following fields:
- id integer, The item’s ID.
- createdAt string,
- updatedAt string,
- name string,
- currentDeploymentId integer,
- lastDeploy object,

services array, An array containing the following fields:
- id integer, The item’s ID.
- createdAt string,
- updatedAt string,
- name string,
- currentDeploymentId integer,
- lastDeploy object,

workflows array, An array containing the following fields:
- id integer, The item’s ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,
- lastExecution object,

reports array, An array containing the following fields:
- id integer, The item’s ID.
- createdAt string,
- updatedAt string,
- name string,
- state string,

**scriptTemplates array**, An array containing the following fields:
- id integer, The item’s ID.
- createdAt string,
- updatedAt string,
- name string,

**files array**, An array containing the following fields:
- id integer, The item’s ID.
- createdAt string,
- updatedAt string,
- fileName string,
- fileSize integer,
- expired boolean,

**enhancements array**, An array containing the following fields:
- id integer, The item’s ID.
- createdAt string,
- updatedAt string,
- name string,
- lastRun object,

**appInstances array**, An array containing the following fields:
- id integer, The item’s ID.
- createdAt string,
- updatedAt string,
- name string,
- slug string,

**projects array**, An array containing the following fields:
- id integer, The item’s ID.
- createdAt string,
- updatedAt string,
- name string,
- description string,

**allObjects array**, An array containing the following fields:
- projectId integer,
- objectId integer,
- objectType string,
- fcoType string,
- subType string,
projects_list

Description
List projects

Usage

projects_list(
    author = NULL,
    permission = NULL,
    hidden = NULL,
    archived = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)

Arguments

author  string optional. If specified, return projects owned by this author. It accepts a comma-separated list of author ids.
permission string optional. A permissions string, one of "read", "write", or "manage". Lists only projects for which the current user has that permission.
hidden boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
archived string optional. The archival status of the requested item(s).
limit integer optional. Number of results to return. Defaults to 20. Maximum allowed is 1000.
projects_list_parent_projects

**Description**

List the Parent Projects an item belongs to

**Usage**

```python
projects_list_parent_projects(id, hidden = NULL)
```
Arguments

id  integer required. The ID of the item.
hidden  boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id  integer, The ID for this project.
author  list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
name  string, The name of this project.
description  string, A description of the project.
users  array, An array containing the following fields:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
autoShare  boolean,
createdAt  string,
updatedAt  string,
archived  string, The archival status of the requested item(s).

projects_list_shares  List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

projects_list_shares(id)

Arguments

id  integer required. The ID of the resource that is shared.
Value

An array containing the following fields:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,

- **writers** list, A list containing the following elements:
  - users array,
  - groups array,

- **owners** list, A list containing the following elements:
  - users array,
  - groups array,

- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**projects_post**  
*Create a project*

---

Description

Create a project

Usage

```
projects_post(name, description, note = NULL, hidden = NULL)
```

Arguments

- **name** string required. The name of this project.
- **description** string required. A description of the project.
- **note** string optional. Notes for the project.
- **hidden** boolean optional. The hidden status of the item.
Value

A list containing the following elements:

- **id** integer, The ID for this project.
- **author** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **name** string, The name of this project.
- **description** string, A description of the project.
- **users** array, An array containing the following fields:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **autoShare** boolean,
- **createdAt** string,
- **updatedAt** string,
- **tables** array, An array containing the following fields:
  - **schema** string,
  - **name** string,
  - **rowCount** integer,
  - **columnCount** integer,
  - **createdAt** string,
  - **updatedAt** string,
- **surveys** array, An array containing the following fields:
  - **id** integer, The item’s ID.
  - **createdAt** string,
  - **updatedAt** string,
- **scripts** array, An array containing the following fields:
  - **id** integer, The item’s ID.
  - **createdAt** string,
  - **updatedAt** string,
  - **name** string,
  - **type** string,
  - **finishedAt** string,
  - **state** string,
• lastRun object,

imports array, An array containing the following fields:
  • id integer, The item’s ID.
  • createdAt string,
  • updatedAt string,
  • name string,
  • type string,
  • finishedAt string,
  • state string,
  • lastRun object,

exports array, An array containing the following fields:
  • id integer, The item’s ID.
  • createdAt string,
  • updatedAt string,
  • name string,
  • type string,
  • finishedAt string,
  • state string,
  • lastRun object,

models array, An array containing the following fields:
  • id integer, The item’s ID.
  • createdAt string,
  • updatedAt string,
  • name string,
  • state string,

notebooks array, An array containing the following fields:
  • id integer, The item’s ID.
  • createdAt string,
  • updatedAt string,
  • name string,
  • currentDeploymentId integer,
  • lastDeploy object,

services array, An array containing the following fields:
  • id integer, The item’s ID.
  • createdAt string,
  • updatedAt string,
  • name string,
  • currentDeploymentId integer,
  • lastDeploy object,

workflows array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• name string,
• state string,
• lastExecution object,

reports array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• name string,
• state string,

scriptTemplates array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• name string,

files array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• fileName string,
• fileSize integer,
• expired boolean,

enhancements array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• name string,
• lastRun object,

appInstances array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• name string,
• slug string,

projects array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
projects_put

- name string,
- description string,

allObjects array, An array containing the following fields:
- projectId integer,
- objectId integer,
- objectType string,
- fcoType string,
- subType string,
- name string,
- icon string,
- author string,
- updatedAt string,
- archived string, The archival status of the requested item(s).
- hidden boolean, The hidden status of the item.

note string,
hidden boolean, The hidden status of the item.
archived string, The archival status of the requested item(s).
parentProject list, A list containing the following elements:
- id integer, The parent project’s ID.
- name integer, The parent project’s name.

projects_put

**Update a project**

**Description**

Update a project

**Usage**

```python
projects_put(
    project_id,
    name = NULL,
    description = NULL,
    note = NULL,
    auto_share = NULL
)
```

**Arguments**

- project_id integer required.
- name string optional. The name of this project.
- description string optional. A description of the project.
- note string optional. Notes for the project.
- auto_share boolean optional. A toggle for sharing the objects within the project when the project is shared. This does not automatically share new objects to the project.
Value

A list containing the following elements:

- **id** integer, The ID for this project.
- **author** list, A list containing the following elements:
  - • **id** integer, The ID of this user.
  - • **name** string, This user’s name.
  - • **username** string, This user’s username.
  - • **initials** string, This user’s initials.
  - • **online** boolean, Whether this user is online.
- **name** string, The name of this project.
- **description** string, A description of the project.
- **users** array, An array containing the following fields:
  - • **id** integer, The ID of this user.
  - • **name** string, This user’s name.
  - • **username** string, This user’s username.
  - • **initials** string, This user’s initials.
  - • **online** boolean, Whether this user is online.
- **autoShare** boolean,
- **createdAt** string,
- **updatedAt** string,
- **tables** array, An array containing the following fields:
  - • **schema** string,
  - • **name** string,
  - • **rowCount** integer,
  - • **columnCount** integer,
  - • **createdAt** string,
  - • **updatedAt** string,
- **surveys** array, An array containing the following fields:
  - • **id** integer, The item’s ID.
  - • **createdAt** string,
  - • **updatedAt** string,
- **scripts** array, An array containing the following fields:
  - • **id** integer, The item’s ID.
  - • **createdAt** string,
  - • **updatedAt** string,
  - • **name** string,
  - • **type** string,
  - • **finishedAt** string,
  - • **state** string,
imports array, An array containing the following fields:
  • id integer, The item’s ID.
  • createdAt string,
  • updatedAt string,
  • name string,
  • type string,
  • finishedAt string,
  • state string,
  • lastRun object,

exports array, An array containing the following fields:
  • id integer, The item’s ID.
  • createdAt string,
  • updatedAt string,
  • name string,
  • type string,
  • finishedAt string,
  • state string,
  • lastRun object,

models array, An array containing the following fields:
  • id integer, The item’s ID.
  • createdAt string,
  • updatedAt string,
  • name string,
  • state string,

notebooks array, An array containing the following fields:
  • id integer, The item’s ID.
  • createdAt string,
  • updatedAt string,
  • name string,
  • currentDeploymentId integer,
  • lastDeploy object,

services array, An array containing the following fields:
  • id integer, The item’s ID.
  • createdAt string,
  • updatedAt string,
  • name string,
  • currentDeploymentId integer,
  • lastDeploy object,

workflows array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• name string,
• state string,
• lastExecution object,

reports array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• name string,
• state string,

scriptTemplates array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• name string,

files array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• fileName string,
• fileSize integer,
• expired boolean,

enhancements array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• name string,
• lastRun object,

appInstances array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• name string,
• slug string,

projects array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
projects_put_archive

- name string,
- description string,

allObjects array, An array containing the following fields:
- projectId integer,
- objectId integer,
- objectType string,
- fcoType string,
- subType string,
- name string,
- icon string,
- author string,
- updatedAt string,
- archived string, The archival status of the requested item(s).
- hidden boolean, The hidden status of the item.

note string,

hidden boolean, The hidden status of the item.

archived string, The archival status of the requested item(s).

parentProject list, A list containing the following elements:
- id integer, The parent project’s ID.
- name integer, The parent project’s name.

projects_put_archive Update the archive status of this object

Description

Update the archive status of this object

Usage

projects_put_archive(id, status)

Arguments

id integer required. The ID of the object.

status boolean required. The desired archived status of the object.
value

A list containing the following elements:

- id (integer): The ID for this project.
- author (list): A list containing the following elements:
  - id (integer): The ID of this user.
  - name (string): This user’s name.
  - username (string): This user’s username.
  - initials (string): This user’s initials.
  - online (boolean): Whether this user is online.
- name (string): The name of this project.
- description (string): A description of the project.
- users (array): An array containing the following fields:
  - id (integer): The ID of this user.
  - name (string): This user’s name.
  - username (string): This user’s username.
  - initials (string): This user’s initials.
  - online (boolean): Whether this user is online.
- autoShare (boolean)
- createdAt (string)
- updatedAt (string)
- tables (array): An array containing the following fields:
  - schema (string)
  - name (string)
  - rowCount (integer)
  - columnCount (integer)
  - createdAt (string)
  - updatedAt (string)
- surveys (array): An array containing the following fields:
  - id (integer): The item’s ID.
  - createdAt (string)
  - updatedAt (string)
- scripts (array): An array containing the following fields:
  - id (integer): The item’s ID.
  - createdAt (string)
  - updatedAt (string)
  - name (string)
  - type (string)
  - finishedAt (string)
  - state (string)
• lastRun object,

imports array, An array containing the following fields:
  • id integer, The item’s ID.
  • createdAt string,
  • updatedAt string,
  • name string,
  • type string,
  • finishedAt string,
  • state string,
  • lastRun object,

exports array, An array containing the following fields:
  • id integer, The item’s ID.
  • createdAt string,
  • updatedAt string,
  • name string,
  • type string,
  • finishedAt string,
  • state string,
  • lastRun object,

models array, An array containing the following fields:
  • id integer, The item’s ID.
  • createdAt string,
  • updatedAt string,
  • name string,
  • state string,

notebooks array, An array containing the following fields:
  • id integer, The item’s ID.
  • createdAt string,
  • updatedAt string,
  • name string,
  • currentDeploymentId integer,
  • lastDeploy object,

services array, An array containing the following fields:
  • id integer, The item’s ID.
  • createdAt string,
  • updatedAt string,
  • name string,
  • currentDeploymentId integer,
  • lastDeploy object,

workflows array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• name string,
• state string,
• lastExecution object,

reports array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• name string,
• state string,

scriptTemplates array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• name string,

files array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• fileName string,
• fileSize integer,
• expired boolean,

enhancements array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• name string,
• lastRun object,

appInstances array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
• name string,
• slug string,

projects array, An array containing the following fields:
• id integer, The item’s ID.
• createdAt string,
• updatedAt string,
**projects_put_parent_projects**

Add an item to a Parent Project

**Description**
Add an item to a Parent Project

**Usage**
```
projects_put_parent_projects(id, parent_project_id)
```

**Arguments**
- **id** integer required. The ID of the item.
- **parent_project_id** integer required. The ID of the Parent Project.

**Value**
An empty HTTP response
projects_put_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

projects_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)

Arguments

id          integer required. The ID of the resource that is shared.
group_ids   array required. An array of one or more group IDs.
permission_level  string required. Options are: "read", "write", or "manage".
share_email_body  string optional. Custom body text for e-mail sent on a share.
send_shared_email  boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers    list, A list containing the following elements:
  • users array,
  • groups array,
writers    list, A list containing the following elements:
  • users array,
  • groups array,
owners     list, A list containing the following elements:
  • users array,
  • groups array,
totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares
  integer. For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

projects_put_shares_users

*Set the permissions users have on this object*

**Description**

Set the permissions users have on this object

**Usage**

```python
projects_put_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```

**Arguments**

- **id**
  integer required. The ID of the resource that is shared.

- **user_ids**
  array required. An array of one or more user IDs.

- **permission_level**
  string required. Options are: "read", "write", or "manage".

- **share_email_body**
  string optional. Custom body text for e-mail sent on a share.

- **send_shared_email**
  boolean optional. Send email to the recipients of a share.

**Value**

A list containing the following elements:

- **readers**
  list, A list containing the following elements:
    - users array,
    - groups array,

- **writers**
  list, A list containing the following elements:
    - users array,
    - groups array,

- **owners**
  list, A list containing the following elements:
    - users array,
- groups array,

totalUserShares
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**publish_addin**

*Add in to Publish an R Markdown file to Platform Reports from RStudio*

**Description**

This function is called when the "Publish To Civis" button is pressed in the RStudio IDE AddIn menu. It will attempt to save the current changes of the active document and push it to the appropriate location in Platform. RMarkdown files will be rendered to html with `publish_rmd` and published as a report. Currently only RMarkdown files are supported.

**Usage**

```r
publish_addin()
```

**See Also**

- `publish_rmd` for publishing RMarkdown files

---

**publish_html**

*Publish HTML to Platform Reports*

**Description**

Publish HTML to Platform Reports

**Usage**

```r
publish_html(
  html_file,
  report_id = NULL,
  report_name = NULL,
  provide_api_key = NULL,
  project_id = NULL
)
```
Publish an R Markdown file to Platform Reports

Description

Publish an R Markdown file to Platform Reports

Usage

```r
publish_rmd(
  rmd_file,
  report_id = NULL,
  report_name = NULL,
  provide_api_key = NULL,
  project_id = NULL,
  ...
)
```

Arguments

- `rmd_file` string, R Markdown file (.Rmd)
- `report_id` integer, ID of an existing report, if provided, the contents of the report will be replaced. If `report_id` is NULL, a new report will be created.
- `report_name` string, Title of report in platform
- `provide_api_key` bool, Set to true to include API key in report.
- `project_id` integer, Project_id that the report should be added to.
provide_api_key
  bool, Set to true to include API key in report.
project_id
  integer, Project_id that the report should be added to.
...
  additional parameters to send to rmarkdown::render. Note: A temporary file will be used for output_file if output_file is not explicitly set and input will be overwritten with rmd_file.

Details
This function also supports passing report_id, report_name, provide_api_key and project_id as metadata in the report's YAML front matter. Just as the title of an RMarkdown document can be set with title: "my title!", these parameters can be set like

```r
civis:
  report_name: "My Report Name"
  report_id: 9000
```

Since report_id is set, this code will overwrite the existing report with that number, which may be useful when updating a report on a schedule. Any argument passed in explicitly to publish_rmd will be used in place of the corresponding argument set in YAML metadata.

Note
rmarkdown::render depends on a recent version of pandoc. pandoc is distributed with RStudio and thus, publish_rmd will work in an RStudio environment. Outside of RStudio pandoc may be installed or knitr::knit2html can be used to convert an R Markdown document into html. The html can then be published to Civis with publish_html.

See Also

- publish_html to publish html to Civis

Examples

```r
## Not run:

# Publish a standard report
publish_rmd("my_beautiful_doc.Rmd")

# Publish a parameterized R Markdown document
# See: http://rmarkdown.rstudio.com/developer_parameterized_reports.html
params <- list("region" = "east", start = as.Date("2015-02-01"))
publish_rmd("my_parameterized_doc.Rmd", params=params)

## End(Not run)
```
**queries_delete_runs**  
*Cancel a run*

**Description**  
Cancel a run

**Usage**  
`queries_delete_runs(id, run_id)`

**Arguments**

- **id**  
  integer required. The ID of the query.
- **run_id**  
  integer required. The ID of the run.

**Value**

An empty HTTP response

---

**queries_get**  
*Get details about a query*

**Description**  
Get details about a query

**Usage**  
`queries_get(id)`

**Arguments**

- **id**  
  integer required. The query ID.

**Value**

A list containing the following elements:

- **id**  
  integer, The query ID.
- **database**  
  integer, The database ID.
- **sql**  
  string, The SQL to execute.
- **credential**  
  integer, The credential ID.
- **resultRows**  
  array, A preview of rows returned by the query.
- **resultColumns**  
  array, A preview of columns returned by the query.
scriptId integer, The ID of the script associated with this query.
exception string, Deprecated and not used.
error string, The error message for this run, if present.
createdAt string,
updatedAt string,
finishedAt string, The end time of the last run.
state string, The state of the last run.
lastRunId integer, The ID of the last run.
hidden boolean, The hidden status of the item.
name string, The name of the query.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
startedAt string, The start time of the last run.
reportId integer, The ID of the report associated with this query.

---

**queries_get_runs**

*Check status of a run*

**Description**

Check status of a run

**Usage**

`queries_get_runs(id, run_id)`

**Arguments**

- **id** integer required. The ID of the query.
- **run_id** integer required. The ID of the run.
queries_list

Value

A list containing the following elements:

- **id**: integer, The ID of the run.
- **queryId**: integer, The ID of the query.
- **state**: string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
- **isCancelRequested**: boolean, True if run cancel requested, else false.
- **startedAt**: string, The time the last run started at.
- **finishedAt**: string, The time the last run completed.
- **error**: string, The error, if any, returned by the run.

queries_list

List

Description

List

Usage

queries_list(
    database_id = NULL,
    author_id = NULL,
    created_before = NULL,
    exclude_results = NULL,
    hidden = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)

Arguments

database_id integer optional. The database ID.
author_id integer optional. The author of the query.
created_before string optional. An upper bound for the creation date of the query.
exclude_results boolean optional. If true, does not return cached query results.
hidden boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
### queries_list_runs

**Description**

List runs for the given query

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>limit</td>
<td>integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.</td>
</tr>
<tr>
<td>page_num</td>
<td>integer optional. Page number of the results to return. Defaults to the first page, 1.</td>
</tr>
<tr>
<td>order</td>
<td>string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at.</td>
</tr>
<tr>
<td>order_dir</td>
<td>string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.</td>
</tr>
<tr>
<td>Value</td>
<td>An array containing the following fields:</td>
</tr>
<tr>
<td>id</td>
<td>integer, The query ID.</td>
</tr>
<tr>
<td>database</td>
<td>integer, The database ID.</td>
</tr>
<tr>
<td>sql</td>
<td>string, The SQL to execute.</td>
</tr>
<tr>
<td>credential</td>
<td>integer, The credential ID.</td>
</tr>
<tr>
<td>resultRows</td>
<td>array, A preview of rows returned by the query.</td>
</tr>
<tr>
<td>resultColumns</td>
<td>array, A preview of columns returned by the query.</td>
</tr>
<tr>
<td>scriptId</td>
<td>integer, The ID of the script associated with this query.</td>
</tr>
<tr>
<td>exception</td>
<td>string, Deprecated and not used.</td>
</tr>
<tr>
<td>error</td>
<td>string, The error message for this run, if present.</td>
</tr>
<tr>
<td>createdAt</td>
<td>string,</td>
</tr>
<tr>
<td>updatedAt</td>
<td>string,</td>
</tr>
<tr>
<td>finishedAt</td>
<td>string, The end time of the last run.</td>
</tr>
<tr>
<td>state</td>
<td>string, The state of the last run.</td>
</tr>
<tr>
<td>lastRunId</td>
<td>integer, The ID of the last run.</td>
</tr>
<tr>
<td>previewRows</td>
<td>integer, The number of rows to save from the query’s result (maximum: 100).</td>
</tr>
<tr>
<td>startedAt</td>
<td>string, The start time of the last run.</td>
</tr>
<tr>
<td>reportId</td>
<td>integer, The ID of the report associated with this query.</td>
</tr>
</tbody>
</table>

**List runs for the given query**
Usage

```python
queries_list_runs(
    id,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```

Arguments

- **id** integer required. The ID of the query.
- **limit** integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
- **page_num** integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order** string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
- **order_dir** string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

- **id** integer, The ID of the run.
- **queryId** integer, The ID of the query.
- **state** string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
- **isCancelRequested** boolean, True if run cancel requested, else false.
- **startedAt** string, The time the last run started at.
- **finishedAt** string, The time the last run completed.
- **error** string, The error, if any, returned by the run.

Get the logs for a run

Get the logs for a run
Usage

queries_list_runs_logs(id, run_id, last_id = NULL, limit = NULL)

Arguments

id    integer required. The ID of the query.
run_id integer required. The ID of the run.
last_id integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted. Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id integer, The ID of the log.
createdAt string, The time the log was created.
message string, The log message.
level string, The level of the log. One of unknown,fatal,error,warn,info,debug.

queries_post  

Execute a query

Description

Execute a query

Usage

queries_post(
    database,
    sql,
    preview_rows,
    credential = NULL,
    hidden = NULL,
    interactive = NULL,
    include_header = NULL,
    compression = NULL,
    column_delimiter = NULL,
    unquoted = NULL,
    filename_prefix = NULL
)

**Arguments**

- **database**: integer required. The database ID.
- **sql**: string required. The SQL to execute.
- **preview_rows**: integer required. The number of rows to save from the query’s result (maximum: 100).
- **credential**: integer optional. The credential ID.
- **hidden**: boolean optional. The hidden status of the item.
- **interactive**: boolean optional. Deprecated and not used.
- **include_header**: boolean optional. Whether the CSV output should include a header row [default: true].
- **compression**: string optional. The type of compression. One of gzip or zip, or none [default: gzip].
- **column_delimiter**: string optional. The delimiter to use. One of comma or tab, or pipe [default: comma].
- **unquoted**: boolean optional. If true, will not quote fields.
- **filename_prefix**: string optional. The output filename prefix.

**Value**

A list containing the following elements:

- **id**: integer, The query ID.
- **database**: integer, The database ID.
- **sql**: string, The SQL to execute.
- **credential**: integer, The credential ID.
- **resultRows**: array, A preview of rows returned by the query.
- **resultColumns**: array, A preview of columns returned by the query.
- **scriptId**: integer, The ID of the script associated with this query.
- **exception**: string, Deprecated and not used.
- **error**: string, The error message for this run, if present.
- **createdAt**: string,
- **updatedAt**: string,
- **finishedAt**: string, The end time of the last run.
- **state**: string, The state of the last run.
- **lastRunId**: integer, The ID of the last run.
- **hidden**: boolean, The hidden status of the item.
- **interactive**: boolean, Deprecated and not used.
- **previewRows**: integer, The number of rows to save from the query’s result (maximum: 100).
queries_post_runs

includeHeader  boolean, Whether the CSV output should include a header row [default: true].
compression   string, The type of compression. One of gzip or zip, or none [default: gzip].
columnDelimiter string, The delimiter to use. One of comma or tab, or pipe [default: comma].
unquoted      boolean, If true, will not quote fields.
filenamePrefix string, The output filename prefix.
startedAt     string, The start time of the last run.
reportId      integer, The ID of the report associated with this query.

| queries_post_runs | Start a run |

Description

Start a run

Usage

queries_post_runs(id)

Arguments

id  integer required. The ID of the query.

Value

A list containing the following elements:

id  integer, The ID of the run.
queryId  integer, The ID of the query.
state  string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested  boolean, True if run cancel requested, else false.
startedAt  string, The time the last run started at.
finishedAt  string, The time the last run completed.
error  string, The error, if any, returned by the run.
queries_put_scripts

**Update the query’s associated script**

**Description**

Update the query’s associated script

**Usage**

`queries_put_scripts(id, script_id)`

**Arguments**

- `id` integer required. The query ID.
- `script_id` integer required. The ID of the script associated with this query.

**Value**

A list containing the following elements:

- `id` integer, The query ID.
- `database` integer, The database ID.
- `sql` string, The SQL to execute.
- `credential` integer, The credential ID.
- `resultRows` array, A preview of rows returned by the query.
- `resultColumns` array, A preview of columns returned by the query.
- `scriptId` integer, The ID of the script associated with this query.
- `exception` string, Deprecated and not used.
- `error` string, The error message for this run, if present.
- `createdAt` string,
- `updatedAt` string,
- `finishedAt` string, The end time of the last run.
- `state` string, The state of the last run.
- `lastRunId` integer, The ID of the last run.
- `hidden` boolean, The hidden status of the item.
- `name` string, The name of the query.
- `author` list, A list containing the following elements:
  - `id` integer, The ID of this user.
  - `name` string, This user’s name.
  - `username` string, This user’s username.
  - `initials` string, This user’s initials.
  - `online` boolean, Whether this user is online.
- `startedAt` string, The start time of the last run.
- `reportId` integer, The ID of the report associated with this query.
query_civis  

Run a Query on Platform

Description

Utility to run queries that return no output.

A default database can be set using options(civis.default_db = "my_database"). If there is only one database available, this database will automatically be used as the default.

Usage

query_civis(x, ...)

## S3 method for class 'sql'
query_civis(x, database = NULL, verbose = FALSE, ...)

## S3 method for class 'numeric'
query_civis(x, verbose = FALSE, ...)

## S3 method for class 'character'
query_civis(x, database = NULL, verbose = FALSE, ...)

Arguments

x  sql("..."), "query", or id of an existing sql script.
...
...  arguments passed to queries_post.
database  string, Name of database where query is run.
verbose  bool, Print detailed updates of job status.

Methods (by class)

- sql: Run a SQL query.
- numeric: Run a SQL query from a previous SQL query id.
- character: Run a SQL query.

See Also

read_civis for downloading results of SQL scripts from Civis Platform as a data frame.

io

Other io: download_civis(), query_civis_file(), read_civis(), write_civis_file(), write_civis()
query_civis_file

Examples

```r
## Not run:
query_civis("GRANT ALL ON schema.my_table TO GROUP admin", "database", credential=0000)

## End(Not run)
```

query_civis_file  Export results from a query to S3 and return a file id.

Description

Exports results from a Redshift SQL query, and returns the id of the file on S3 for use with read_civis or download_civis.

Usage

```r
query_civis_file(x, ...)
```

```
# S3 method for class 'character'
query_civis_file(
  x,
  database = NULL,
  job_name = NULL,
  hidden = TRUE,
  verbose = verbose,
  csv_settings = NULL,
  ...
)
```

```
# S3 method for class 'sql'
query_civis_file(
  x,
  database = NULL,
  job_name = NULL,
  hidden = TRUE,
  verbose = FALSE,
  csv_settings = NULL,
  ...
)
```

```
# S3 method for class 'numeric'
query_civis_file(x, database = NULL, verbose = FALSE, ...)
```
Arguments

- **x**: "schema.table", sql("query"), or a sql script job id.
- **...**: Options passed to scripts_post_sql, including credential.
- **database**: string, Name of database where data frame is to be uploaded. If no database is specified, uses options(civis.default_db).
- **job_name**: string, Name of the job (default: "Civis S3 Export Via R Client").
- **hidden**: bool, Whether the job is hidden.
- **verbose**: bool, Set to TRUE to print intermediate progress indicators.
- **csv_settings**: See scripts_post_sql for details.

Details

By default, the export uses the default csv_settings in scripts_post_sql, which is a gzipped csv.

Methods (by class)

- **character**: Export a "schema.table" to a file id.
- **sql**: Export results of a query to a file id.
- **numeric**: Run an existing sql script and return the file id of the results on S3.

See Also

Other io: download_civis(), query_civis(), read_civis(), write_civis_file(), write_civis()

Examples

```r
## Not run:
id <- query_civis_file("schema.tablename", database = "my_database")
df <- read_civis(id, using = read.csv)

query <- sql("SELECT * FROM table JOIN other_table USING id WHERE var1 < 23")
id <- query_civis_file(query)
df <- read_civis(id, using = read.csv)

id <- query_civis_file(query_id, credential_id = 0000)
df <- read_civis(id, using = read.csv)

## End(Not run)
```
**read_civis**

*Read tables and files from Civis Platform*

**Description**

`read_civis` loads a table from Redshift as a data frame if given a "schema.table" or `sql("query")` as the first argument, or loads a file from Amazon S3 (the files endpoint) if a file id is given. Run outputs from any Civis platform script are returned if a `civis_script` is given.

A default database can be set using `options(civis.default_db = "my_database")`. If there is only one database available, this database will automatically be used as the default.

**Usage**

```r
read_civis(x, ...)

## S3 method for class 'numeric'
read_civis(x, using = read.csv, verbose = FALSE, ...)
```

```r
## S3 method for class 'character'
read_civis(x, database = NULL, ...)
```

```r
## S3 method for class 'sql'
read_civis(
  x,
  database = NULL,
  using = utils::read.csv,
  job_name = NULL,
  hidden = TRUE,
  verbose = FALSE,
  ...
)
```

```r
## S3 method for class 'civis_script'
read_civis(x, using, regex = NULL, ...)
```

**Arguments**

- `x` "schema.table", `sql("query")`, or a file id.
- `...` arguments passed to `using`.
- `using` function, Function to convert the file to a data frame or to unserialize. the file (e.g. `read.csv` or `readRDS`).
- `verbose` bool, Set to TRUE to print intermediate progress indicators.
- `database` string, Name of database where data frame is to be uploaded. If no database is specified, uses `options(civis.default_db)`.
- `job_name` string, Name of the job (default: "Civis Export Via R Client").
hidden      bool, Whether the job is hidden.
regex       Regex of matching run output names.

Details
By default, `read_civis.numeric` assumes the file is a CSV. For reading a serialized R object, set
`using = readRDS` for example.
If `using = NULL`, `read_civis.civis_script` will return all JSONValues with name matching
regex. Otherwise all File run outputs matching regex will be read into memory with using. Re-
results are always a named list. If the script has no outputs, an empty list will be returned.

Methods (by class)
- numeric: Return a file as a data frame
- character: Return all columns from a table as a data frame.
- sql: Return a SQL query as a data frame.
- civis_script: Return run outputs of a civis_script as a named list.

See Also
Other io: `download_civis()`, `query_civis_file()`, `query_civis()`, `write_civis_file()`, `write_civis()`

Examples
```r
## Not run:
# Read all columns in a single table
df <- read_civis("schema.my_table", database = "my_database")

# Read data from a SQL select statement
query <- sql("SELECT * FROM table JOIN other_table USING id WHERE var1 < 23")
df <- read_civis(query, database = "my_database")

# Read an R object from the files endpoint.
id <- write_civis_file(df)
df <- read_civis(id)

# Read a text file or csv from the files endpoint.
id <- write_civis_file("my_csv.csv")
df <- read_civis(id)

# Read JSONValues from a civis script
vals <- read_civis(civis_script(1234))

# Read File run outputs from a civis script
df <- read_civis(civis_script(1234), regex = ".csv", using = read.csv)
obj <- read_civis(civis_script(1234), regex = ".rds", using = readRDS)

# Gracefully handle when read_civis.sql returns no rows
query <- sql("SELECT * FROM table WHERE 1 = 2")
mean_x <- tryCatch({
```

df <- read_civis(query, database = "my_database")
mean(df$x)
}, empty_result_error = function(e) {
  NA
})

## End(Not run)

---

**refresh_table**

*Refresh a table*

**Description**

Refreshes a table on Redshift using `tables_post_refresh`, which runs the table scanner and updates table meta-data.

**Usage**

```r
refresh_table(tablename, database = NULL, verbose = FALSE)
```

**Arguments**

- **tablename** string, Name of table and schema "schema.tablename"
- **database** string, Name of database where data frame is to be uploaded. If no database is specified, uses `options(civis.default_db)`
- **verbose** bool, Set to TRUE to print intermediate progress indicators.

**Value**

Returns table meta-data from `tables_get`.

**See Also**

Other tables: `get_table_id()`, `transfer_table()`

---

**remote_hosts_list**

*List the remote hosts*

**Description**

List the remote hosts

**Usage**

```r
remote_hosts_list(type = NULL)
```
Arguments

type string optional. The type of remote host. One of: RemoteHostTypes::Bitbucket, RemoteHostTypes::GitSSH, RemoteHostTypes::Github, RemoteHostTypes::GoogleDoc, RemoteHostTypes::JDBC, RemoteHostTypes::Postgres, RemoteHostTypes::Redshift, RemoteHostTypes::S3Storage, and RemoteHostTypes::Salesforce

Value

An array containing the following fields:

id integer, The ID of the remote host.
name string, The name of the remote host.
type string. The type of remote host. One of: RemoteHostTypes::Bitbucket, RemoteHostTypes::GitSSH, RemoteHostTypes::Github, RemoteHostTypes::GoogleDoc, RemoteHostTypes::JDBC, RemoteHostTypes::Postgres, RemoteHostTypes::Redshift, RemoteHostTypes::S3Storage, and RemoteHostTypes::Salesforce
url string, The URL for remote host.

remote_hosts_list_data_sets

List data sets available from a remote host

Description

List data sets available from a remote host

Usage

remote_hosts_list_data_sets(
    id,
    credential_id = NULL,
    username = NULL,
    password = NULL,
    q = NULL,
    s = NULL
)

Arguments

id integer required. The ID of the remote host.
credential_id integer optional. The credential ID.
username string optional. The user name for remote host.
password string optional. The password for remote host.
q string optional. The query string for data set.
s boolean optional. If true will only return schemas, otherwise, the results will be the full path.
**remote_hosts_post**

**Value**

An array containing the following fields:

- **name** string, The path to a data_set.
- **fullPath** boolean, Boolean that indicates whether further querying needs to be done before the table can be selected.

---

**remote_hosts_post** Create a new remote host

**Description**

Create a new remote host

**Usage**

remote_hosts_post(name, url, type)

**Arguments**

- **name** string required. The human readable name for the remote host.
- **url** string required. The URL to your host.
- **type** string required. The type of remote host. One of: RemoteHostTypes::Bitbucket, RemoteHostTypes::GitSSH, RemoteHostTypes::Github, RemoteHostTypes::GoogleDoc, RemoteHostTypes::JDBC, RemoteHostTypes::Postgres, RemoteHostTypes::Redshift, RemoteHostTypes::S3Storage, and RemoteHostTypes::Salesforce

**Value**

A list containing the following elements:

- **id** integer, The ID of the remote host.
- **name** string, The name of the remote host.
- **type** string, The type of remote host. One of: RemoteHostTypes::Bitbucket, RemoteHostTypes::GitSSH, RemoteHostTypes::Github, RemoteHostTypes::GoogleDoc, RemoteHostTypes::JDBC, RemoteHostTypes::Postgres, RemoteHostTypes::Redshift, RemoteHostTypes::S3Storage, and RemoteHostTypes::Salesforce
- **url** string, The URL for remote host.
remote_hosts_post_authenticate

Authenticate against a remote host using either a credential or a user name and password

Description

Authenticate against a remote host using either a credential or a user name and password

Usage

remote_hosts_post_authenticate(
    id,
    credential_id = NULL,
    username = NULL,
    password = NULL
)

Arguments

id integer required. The ID of the remote host.
credential_id integer optional. The credential ID.
username string optional. The user name for remote host.
password string optional. The password for remote host.

Value

An empty HTTP response

reports_delete_grants

Revoke permission for this report to perform Civis platform API operations on your behalf

Description

Revoke permission for this report to perform Civis platform API operations on your behalf

Usage

reports_delete_grants(id)

Arguments

id integer required. The ID of this report.
Value

An empty HTTP response

---

reports_delete_projects

Remove a Report from a project

Description

Remove a Report from a project

Usage

reports_delete_projects(id, project_id)

Arguments

id integer required. The ID of the Report.
project_id integer required. The ID of the project.

Value

An empty HTTP response

---

reports_delete_services_projects

Remove a Service Report from a project

Description

Remove a Service Report from a project

Usage

reports_delete_services_projects(id, project_id)

Arguments

id integer required. The ID of the Service Report.
project_id integer required. The ID of the project.

Value

An empty HTTP response
### reports_delete_services_shares_groups

Revoke the permissions a group has on this object

**Description**

Revoke the permissions a group has on this object

**Usage**

```
reports_delete_services_shares_groups(id, group_id)
```

**Arguments**

- **id**: integer required. The ID of the resource that is shared.
- **group_id**: integer required. The ID of the group.

**Value**

An empty HTTP response

### reports_delete_services_shares_users

Revoke the permissions a user has on this object

**Description**

Revoke the permissions a user has on this object

**Usage**

```
reports_delete_services_shares_users(id, user_id)
```

**Arguments**

- **id**: integer required. The ID of the resource that is shared.
- **user_id**: integer required. The ID of the user.

**Value**

An empty HTTP response
reports_delete_shares_groups
\[\text{Revoke the permissions a group has on this object}\]

**Description**
Revoke the permissions a group has on this object

**Usage**
\[\text{reports_delete_shares_groups(id, group_id)}\]

**Arguments**
- **id** integer required. The ID of the resource that is shared.
- **group_id** integer required. The ID of the group.

**Value**
An empty HTTP response

reports_delete_shares_users
\[\text{Revoke the permissions a user has on this object}\]

**Description**
Revoke the permissions a user has on this object

**Usage**
\[\text{reports_delete_shares_users(id, user_id)}\]

**Arguments**
- **id** integer required. The ID of the resource that is shared.
- **user_id** integer required. The ID of the user.

**Value**
An empty HTTP response
reports_get  

*Show a single report*

**Description**

Show a single report

**Usage**

`reports_get(id)`

**Arguments**

| id    | integer required. The ID of this report. |

**Value**

A list containing the following elements:

<p>| id  | integer, The ID of this report. |
| name | string, The name of the report. |
| user | list, A list containing the following elements: |
|      | • id integer, The ID of this user. |
|      | • name string, This user’s name. |
|      | • username string, This user’s username. |
|      | • initials string, This user’s initials. |
|      | • online boolean, Whether this user is online. |
| createdAt | string, |
| updatedAt | string, |
| projects | array, An array containing the following fields: |
|      | • id integer, The ID for the project. |
|      | • name string, The name of the project. |
| state | string, The status of the report’s last run. |
| finishedAt | string, The time that the report’s last run finished. |
| vizUpdatedAt | string, The time that the report’s visualization was last updated. |
| script | list, A list containing the following elements: |
|      | • id integer, The ID for the script. |
|      | • name string, The name of the script. |
|      | • sql string, The raw SQL query for the script. |
| jobPath | string, The link to details of the job that backs this report. |
| tableauId | integer, |
| type | string, |</p>
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>templateId</td>
<td>integer, The ID of the template used for this report.</td>
</tr>
<tr>
<td>authThumbnailUrl</td>
<td>string, URL for a thumbnail of the report.</td>
</tr>
<tr>
<td>lastRun</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• id integer,</td>
</tr>
<tr>
<td></td>
<td>• state string,</td>
</tr>
<tr>
<td></td>
<td>• createdAt string, The time that the run was queued.</td>
</tr>
<tr>
<td></td>
<td>• startedAt string, The time that the run started.</td>
</tr>
<tr>
<td></td>
<td>• finishedAt string, The time that the run completed.</td>
</tr>
<tr>
<td></td>
<td>• error string, The error message for this run, if present.</td>
</tr>
<tr>
<td>archived</td>
<td>string, The archival status of the requested item(s).</td>
</tr>
<tr>
<td>hidden</td>
<td>boolean, The hidden status of the item.</td>
</tr>
<tr>
<td>authDataUrl</td>
<td>string,</td>
</tr>
<tr>
<td>authCodeUrl</td>
<td>string,</td>
</tr>
<tr>
<td>config</td>
<td>string, Any configuration metadata for this report.</td>
</tr>
<tr>
<td>validOutputFile</td>
<td>boolean, Whether the job (a script or a query) that backs the report currently has a valid output file.</td>
</tr>
<tr>
<td>provideAPIKey</td>
<td>boolean, Whether the report requests an API Key from the report viewer.</td>
</tr>
<tr>
<td>apiKey</td>
<td>string, A Civis API key that can be used by this report.</td>
</tr>
<tr>
<td>apiKeyId</td>
<td>integer, The ID of the API key. Can be used for auditing API use by this report.</td>
</tr>
<tr>
<td>appState</td>
<td>list, Any application state blob for this report.</td>
</tr>
<tr>
<td>useViewersTableauUsername</td>
<td>boolean, Apply user level filtering on Tableau reports.</td>
</tr>
</tbody>
</table>

---

**reports_get_git_commits**

*Get file contents at commit_hash*

**Description**

Get file contents at commit_hash

**Usage**

```plaintext
reports_get_git_commits(id, commit_hash)
```

**Arguments**

- **id**
  - integer required. The ID of the file.
- **commit_hash**
  - string required. The SHA (full or shortened) of the desired git commit.
Value
A list containing the following elements:

- `content` (string): The file’s contents.
- `type` (string): The file’s type.
- `size` (integer): The file’s size.
- `fileHash` (string): The SHA of the file.

---

**reports_get_services**  *Show a single service report*

---

**Description**
Show a single service report

**Usage**

```
reports_get_services(id)
```

**Arguments**

- `id` (integer, required): The ID of this report.

**Value**
A list containing the following elements:

- `id` (integer): The ID of this report.
- `name` (string): The name of the report.
- `user` (list): A list containing the following elements:
  - `id` (integer): The ID of this user.
  - `name` (string): This user’s name.
  - `username` (string): This user’s username.
  - `initials` (string): This user’s initials.
  - `online` (boolean): Whether this user is online.
- `createdAt` (string)
- `updatedAt` (string)
- `host` (string): The host for the service report
- `displayUrl` (string): The URL to display the service report.
- `serviceId` (integer): The id of the backing service
- `provideAPIKey` (boolean): Whether the report requests an API Key from the report viewer.
- `apiKey` (string): A Civis API key that can be used by this report.
- `apiKeyId` (integer): The ID of the API key. Can be used for auditing API use by this report.
Description
List Reports

Usage

```python
reports_list(
    type = NULL,
    author = NULL,
    template_id = NULL,
    hidden = NULL,
    archived = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```

Arguments

- **type**: string optional. If specified, return report of these types. It accepts a comma-separated list, possible values are 'tableau' or 'other'.
- **author**: string optional. If specified, return reports from this author. It accepts a comma-separated list of author ids.
- **template_id**: integer optional. If specified, return reports using the provided Template.
- **hidden**: boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
- **archived**: string optional. The archival status of the requested item(s).
- **limit**: integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
- **page_num**: integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order**: string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at.
- **order_dir**: string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value
An array containing the following fields:

- **id**: integer, The ID of this report.
name string, The name of the report.

user list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

createdAt string,

updatedAt string,

projects array, An array containing the following fields:
  • id integer, The ID for the project.
  • name string, The name of the project.

state string, The status of the report’s last run.

finishedAt string, The time that the report’s last run finished.

vizUpdatedAt string, The time that the report’s visualization was last updated.

script list, A list containing the following elements:
  • id integer, The ID for the script.
  • name string, The name of the script.
  • sql string, The raw SQL query for the script.

jobPath string, The link to details of the job that backs this report.

tableauId integer,

type string,

templateId integer, The ID of the template used for this report.

authThumbnailUrl string, URL for a thumbnail of the report.

lastRun list, A list containing the following elements:
  • id integer,
  • state string,
  • createdAt string, The time that the run was queued.
  • startedAt string, The time that the run started.
  • finishedAt string, The time that the run completed.
  • error string, The error message for this run, if present.

archived string, The archival status of the requested item(s).
**reports_list_git**

Get the git metadata attached to an item

**Description**

Get the git metadata attached to an item

**Usage**

`reports_list_git(id)`

**Arguments**

- **id**
  - integer required. The ID of the file.

**Value**

A list containing the following elements:

- `gitRef` string, A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
- `gitBranch` string, The git branch that the file is on.
- `gitPath` string, The path of the file in the repository.
- `gitRepo` list, A list containing the following elements:
  - `id` integer, The ID for this git repository.
  - `repoUrl` string, The URL for this git repository.
  - `createdAt` string,
  - `updatedAt` string.
- `pullFromGit` boolean, Automatically pull latest commit from git. Only works for scripts.

**reports_list_git_commits**

Get the git commits for an item

**Description**

Get the git commits for an item

**Usage**

`reports_list_git_commits(id)`

**Arguments**

- **id**
  - integer required. The ID of the file.
Value

A list containing the following elements:

- **commitHash** string, The SHA of the commit.
- **authorName** string, The name of the commit’s author.
- **date** string, The commit’s timestamp.
- **message** string, The commit message.

Description

List the projects a Report belongs to

Usage

reports_list_projects(id, hidden = NULL)

Arguments

- **id** integer required. The ID of the Report.
- **hidden** boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

- **id** integer, The ID for this project.
- **author** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **name** string, The name of this project.
- **description** string, A description of the project.
- **users** array, An array containing the following fields:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
• online boolean, Whether this user is online.

autoShare boolean,  
createdAt string,  
updatedAt string,  
archived string, The archival status of the requested item(s).

---

reports_list_services_projects

*List the projects a Service Report belongs to*

**Description**

List the projects a Service Report belongs to

**Usage**

`reports_list_services_projects(id, hidden = NULL)`

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of the Service Report.</td>
</tr>
<tr>
<td>hidden</td>
<td>boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.</td>
</tr>
</tbody>
</table>

**Value**

An array containing the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer, The ID for this project.</td>
</tr>
<tr>
<td>author</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• id integer, The ID of this user.</td>
</tr>
<tr>
<td></td>
<td>• name string, This user’s name.</td>
</tr>
<tr>
<td></td>
<td>• username string, This user’s username.</td>
</tr>
<tr>
<td></td>
<td>• initials string, This user’s initials.</td>
</tr>
<tr>
<td></td>
<td>• online boolean, Whether this user is online.</td>
</tr>
<tr>
<td>name</td>
<td>string, The name of this project.</td>
</tr>
<tr>
<td>description</td>
<td>string, A description of the project.</td>
</tr>
<tr>
<td>users</td>
<td>array, An array containing the following fields:</td>
</tr>
<tr>
<td></td>
<td>• id integer, The ID of this user.</td>
</tr>
<tr>
<td></td>
<td>• name string, This user’s name.</td>
</tr>
<tr>
<td></td>
<td>• username string, This user’s username.</td>
</tr>
<tr>
<td></td>
<td>• initials string, This user’s initials.</td>
</tr>
<tr>
<td></td>
<td>• online boolean, Whether this user is online.</td>
</tr>
</tbody>
</table>
autoShare: boolean,
createdAt: string,
updatedAt: string,
archived: string, The archival status of the requested item(s).

reports_list_services_shares

*List users and groups permissioned on this object*

**Description**

List users and groups permissioned on this object

**Usage**

`reports_list_services_shares(id)`

**Arguments**

`id`: integer required. The ID of the resource that is shared.

**Value**

An array containing the following fields:

- **readers**: list, A list containing the following elements:
  - users array,
  - groups array,

- **writers**: list, A list containing the following elements:
  - users array,
  - groups array,

- **owners**: list, A list containing the following elements:
  - users array,
  - groups array,

- **totalUserShares**: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

- **totalGroupShares**: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.
reports_list_shares  

List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

reports_list_shares(id)

Arguments

id  integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers  list, A list containing the following elements:
  • users array,
  • groups array,

writers list, A list containing the following elements:
  • users array,
  • groups array,

owners list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.
reports_patch  

*Update a report*

**Description**

Update a report

**Usage**

```r
reports_patch(
  id,
  name = NULL,
  script_id = NULL,
  code_body = NULL,
  config = NULL,
  app_state = NULL,
  provide_api_key = NULL,
  template_id = NULL,
  use_viewers_tableau_username = NULL
)
```

**Arguments**

- **id**  
  integer required. The ID of the report to modify.
- **name**  
  string optional. The name of the report.
- **script_id**  
  integer optional. The ID of the job (a script or a query) used to create this report.
- **code_body**  
  string optional. The code for the report visualization.
- **config**  
  string optional.
- **app_state**  
  list optional. The application state blob for this report.
- **provide_api_key**  
  boolean optional. Allow the report to provide an API key to front-end code.
- **template_id**  
  integer optional. The ID of the template used for this report. If null is passed, no template will back this report. Changes to the backing template will reset the report appState.
- **use_viewers_tableau_username**  
  boolean optional. Apply user level filtering on Tableau reports.

**Value**

A list containing the following elements:

- **id**  
  integer, The ID of this report.
- **name**  
  string, The name of the report.
- **user**  
  list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

createdAt string,
updatedAt string,
projects array, An array containing the following fields:
  • id integer, The ID for the project.
  • name string, The name of the project.
state string, The status of the report’s last run.
finishedAt string, The time that the report’s last run finished.
vizUpdatedAt string, The time that the report’s visualization was last updated.

script list, A list containing the following elements:
  • id integer, The ID for the script.
  • name string, The name of the script.
  • sql string, The raw SQL query for the script.

jobPath string, The link to details of the job that backs this report.
tableauId integer,
type string,
templateId integer, The ID of the template used for this report.
authThumbnailUrl string, URL for a thumbnail of the report.

lastRun list, A list containing the following elements:
  • id integer,
  • state string,
  • createdAt string, The time that the run was queued.
  • startedAt string, The time that the run started.
  • finishedAt string, The time that the run completed.
  • error string, The error message for this run, if present.

archived string, The archival status of the requested item(s).
hidden boolean, The hidden status of the item.
authDataUrl string,
authCodeUrl string,
config string, Any configuration metadata for this report.
validOutputFile boolean, Whether the job (a script or a query) that backs the report currently has a valid output file.

provideAPIKey boolean, Whether the report requests an API Key from the report viewer.
apiKey string, A Civis API key that can be used by this report.
apiKeyId integer, The ID of the API key. Can be used for auditing API use by this report.
appState list, Any application state blob for this report.
useViewersTableauUsername boolean, Apply user level filtering on Tableau reports.

reports_patch_services

Update some attributes of this service report

Description

Update some attributes of this service report

Usage

reports_patch_services(id, name = NULL, provide_api_key = NULL)

Arguments

id integer required. The ID of this report.
name string optional. The name of the service report.
provide_api_key boolean optional. Whether the report requests an API Key from the report viewer.

Value

A list containing the following elements:

id integer, The ID of this report.
name string, The name of the report.
user list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
createdAt string,
updatedAt string,
host string, The host for the service report
displayUrl string, The URL to display the service report.
serviceId integer, The id of the backing service
provideAPIKey boolean, Whether the report requests an API Key from the report viewer.
apiKey string, A Civis API key that can be used by this report.
apiKeyId integer, The ID of the API key. Can be used for auditing API use by this report.
Create a report

Usage

reports_post(
    script_id = NULL,
    name = NULL,
    code_body = NULL,
    app_state = NULL,
    provide_api_key = NULL,
    template_id = NULL,
    hidden = NULL
)

Arguments

- **script_id**: integer optional. The ID of the job (a script or a query) used to create this report.
- **name**: string optional. The name of the report.
- **code_body**: string optional. The code for the report visualization.
- **app_state**: list optional. Any application state blob for this report.
- **provide_api_key**: boolean optional. Allow the report to provide an API key to front-end code.
- **template_id**: integer optional. The ID of the template used for this report.
- **hidden**: boolean optional. The hidden status of the item.

Value

A list containing the following elements:

- **id**: integer, The ID of this report.
- **name**: string, The name of the report.
- **user**: list, A list containing the following elements:
  - **id**: integer, The ID of this user.
  - **name**: string, This user’s name.
  - **username**: string, This user’s username.
  - **initials**: string, This user’s initials.
  - **online**: boolean, Whether this user is online.
- **createdAt**: string,
- **updatedAt**: string,
projects array, An array containing the following fields:
  • id integer, The ID for the project.
  • name string, The name of the project.
state string, The status of the report’s last run.
finishedAt string, The time that the report’s last run finished.
vizUpdatedAt string, The time that the report’s visualization was last updated.
script list, A list containing the following elements:
  • id integer, The ID for the script.
  • name string, The name of the script.
  • sql string, The raw SQL query for the script.
jobPath string, The link to details of the job that backs this report.
tableauId integer,
type string,
templateId integer, The ID of the template used for this report.
aUTHThumbnailUrl string, URL for a thumbnail of the report.
lastRun list, A list containing the following elements:
  • id integer,
  • state string,
  • createdAt string, The time that the run was queued.
  • startedAt string, The time that the run started.
  • finishedAt string, The time that the run completed.
  • error string, The error message for this run, if present.
archived string, The archival status of the requested item(s).
hidden boolean, The hidden status of the item.
aUTHDataUrl string,
aUTHCodeUrl string,
cONFIG string, Any configuration metadata for this report.
validOutputFile boolean, Whether the job (a script or a query) that backs the report currently has a valid output file.
proVIDeAPIKey boolean, Whether the report requests an API Key from the report viewer.
apiKey string, A Civis API key that can be used by this report.
apiKeyId integer, The ID of the API key. Can be used for auditing API use by this report.
appState list, Any application state blob for this report.
useViewerSTableauUsername boolean, Apply user level filtering on Tableau reports.
**reports_post_git_commits**

*Commit and push a new version of the file*

**Description**

Commit and push a new version of the file

**Usage**

```python
reports_post_git_commits(id, content, message, file_hash)
```

**Arguments**

- **id** integer required. The ID of the file.
- **content** string required. The contents to commit to the file.
- **message** string required. A commit message describing the changes being made.
- **file_hash** string required. The full SHA of the file being replaced.

**Value**

A list containing the following elements:

- **content** string, The file’s contents.
- **type** string, The file’s type.
- **size** integer, The file’s size.
- **fileHash** string, The SHA of the file.

**reports_post_grants**

*Grant this report the ability to perform Civis platform API operations on your behalf*

**Description**

Grant this report the ability to perform Civis platform API operations on your behalf

**Usage**

```python
reports_post_grants(id)
```

**Arguments**

- **id** integer required. The ID of this report.
Value

A list containing the following elements:

- **id** integer, The ID of this report.
- **name** string, The name of the report.
- **user** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **createdAt** string,
- **updatedAt** string,
- **projects** array, An array containing the following fields:
  - **id** integer, The ID for the project.
  - **name** string, The name of the project.
- **state** string, The status of the report’s last run.
- **finishedAt** string, The time that the report’s last run finished.
- **vizUpdatedAt** string, The time that the report’s visualization was last updated.
- **script** list, A list containing the following elements:
  - **id** integer, The ID for the script.
  - **name** string, The name of the script.
  - **sql** string, The raw SQL query for the script.
- **jobPath** string, The link to details of the job that backs this report.
- **tableauId** integer,
- **type** string,
- **templateId** integer, The ID of the template used for this report.
- **authThumbnailUrl** string, URL for a thumbnail of the report.
- **lastRun** list, A list containing the following elements:
  - **id** integer,
  - **state** string,
  - **createdAt** string, The time that the run was queued.
  - **startedAt** string, The time that the run started.
  - **finishedAt** string, The time that the run completed.
  - **error** string, The error message for this run, if present.
- **archived** string, The archival status of the requested item(s).
- **hidden** boolean, The hidden status of the item.
- **authDataUrl** string,
authCodeUrl: string,
config: string, Any configuration metadata for this report.
validOutputFile: boolean, Whether the job (a script or a query) that backs the report currently has a valid output file.
provideAPIKey: boolean, Whether the report requests an API Key from the report viewer.
apiKey: string, A Civis API key that can be used by this report.
apiKeyId: integer, The ID of the API key. Can be used for auditing API use by this report.
apState: list, Any application state blob for this report.
useViewersTableauUsername: boolean, Apply user level filtering on Tableau reports.

reports_post_refresh

Refresh the data in this Tableau report

Description
Refresh the the data in this Tableau report

Usage

reports_post_refresh(id)

Arguments

id: integer required. The ID of this report.

Value

A list containing the following elements:

id: integer, The ID of this report.
organization: list, A list containing the following elements:
  • id: integer, The ID of this organization.
  • tableauRefreshUsage: integer, The number of tableau refreshes used this month.
  • tableauRefreshLimit: integer, The number of monthly tableau refreshes permitted to this organization.
  • tableauRefreshHistory: array, The number of tableau refreshes used this month.
**reports_post_services**  *Create a service report*

**Description**

Create a service report

**Usage**

```ruby
reports_post_services(service_id, provide_api_key = NULL)
```

**Arguments**

- **service_id**  
  integer required. The id of the backing service
- **provide_api_key**  
  boolean optional. Whether the report requests an API Key from the report viewer.

**Value**

A list containing the following elements:

- **id**  
  integer, The ID of this report.
- **name**  
  string, The name of the report.
- **user**  
  list, A list containing the following elements:
  - **id**  
    integer, The ID of this user.
  - **name**  
    string, This user’s name.
  - **username**  
    string, This user’s username.
  - **initials**  
    string, This user’s initials.
  - **online**  
    boolean, Whether this user is online.
- **createdAt**  
  string,
- **updatedAt**  
  string,
- **host**  
  string, The host for the service report
- **displayUrl**  
  string, The URL to display the service report.
- **serviceId**  
  integer, The id of the backing service
- **provideAPIKey**  
  boolean, Whether the report requests an API Key from the report viewer.
- **apiKey**  
  string, A Civis API key that can be used by this report.
- **apiKeyId**  
  integer, The ID of the API key. Can be used for auditing API use by this report.
**Description**

Update the archive status of this object

**Usage**

`reports_put_archive(id, status)`

**Arguments**

- **id** integer required. The ID of the object.
- **status** boolean required. The desired archived status of the object.

**Value**

A list containing the following elements:

- **id** integer, The ID of this report.
- **name** string, The name of the report.
- **user** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **createdAt** string,
- **updatedAt** string,
- **projects** array, An array containing the following fields:
  - **id** integer, The ID for the project.
  - **name** string, The name of the project.
- **state** string, The status of the report’s last run.
- **finishedAt** string, The time that the report’s last run finished.
- **vizUpdatedAt** string, The time that the report’s visualization was last updated.
- **script** list, A list containing the following elements:
  - **id** integer, The ID for the script.
  - **name** string, The name of the script.
  - **sql** string, The raw SQL query for the script.
- **jobPath** string, The link to details of the job that backs this report.
- **tableauId** integer,
type string,
templateId integer, The ID of the template used for this report.
authThumbnailUrl string, URL for a thumbnail of the report.
lastRun list, A list containing the following elements:
  • id integer,
  • state string,
  • createdAt string, The time that the run was queued.
  • startedAt string, The time that the run started.
  • finishedAt string, The time that the run completed.
  • error string, The error message for this run, if present.
archived string, The archival status of the requested item(s).
hidden boolean, The hidden status of the item.
authDataUrl string,
authCodeUrl string,
config string, Any configuration metadata for this report.
validOutputFile boolean, Whether the job (a script or a query) that backs the report currently has
  a valid output file.
provideAPIKey boolean, Whether the report requests an API Key from the report viewer.
apiKey string, A Civis API key that can be used by this report.
apiKeyId integer, The ID of the API key. Can be used for auditing API use by this report.
appState list, Any application state blob for this report.
useViewersTableauUsername boolean, Apply user level filtering on Tableau reports.

---

**reports_put_git**  
*Attach an item to a file in a git repo*

**Description**

Attach an item to a file in a git repo

**Usage**

```python
reports_put_git(
    id,
    git_ref = NULL,
    git_branch = NULL,
    git_path = NULL,
    git_repo_url = NULL,
    pull_from_git = NULL
)
```
**Arguments**

- **id** integer required. The ID of the file.
- **git_ref** string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
- **git_branch** string optional. The git branch that the file is on.
- **git_path** string optional. The path of the file in the repository.
- **git_repo_url** string optional. The URL of the git repository.
- **pull_from_git** boolean optional. Automatically pull latest commit from git. Only works for scripts.

**Value**

A list containing the following elements:

- **gitRef** string, A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
- **gitBranch** string, The git branch that the file is on.
- **gitPath** string, The path of the file in the repository.
- **gitRepo** list, A list containing the following elements:
  - id integer, The ID for this git repository.
  - repoUrl string, The URL for this git repository.
  - createdAt string,
  - updatedAt string,
- **pullFromGit** boolean, Automatically pull latest commit from git. Only works for scripts.

---

**reports_put_projects Add a Report to a project**

**Description**

Add a Report to a project

**Usage**

`reports_put_projects(id, project_id)`

**Arguments**

- **id** integer required. The ID of the Report.
- **project_id** integer required. The ID of the project.

**Value**

An empty HTTP response
**reports_put_services_projects**

*Add a Service Report to a project*

**Description**

Add a Service Report to a project

**Usage**

```
reports_put_services_projects(id, project_id)
```

**Arguments**

- `id` integer required. The ID of the Service Report.
- `project_id` integer required. The ID of the project.

**Value**

An empty HTTP response

---

**reports_put_services_shares_groups**

*Set the permissions groups has on this object*

**Description**

Set the permissions groups has on this object

**Usage**

```
reports_put_services_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```
Arguments

id integer required. The ID of the resource that is shared.
group_ids array required. An array of one or more group IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers list, A list containing the following elements:
  • users array,
  • groups array,

writers list, A list containing the following elements:
  • users array,
  • groups array,

owners list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

Description

Set the permissions users have on this object

Usage

```java
reports_put_services_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```
Arguments

id integer required. The ID of the resource that is shared.
user_ids array required. An array of one or more user IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers list, A list containing the following elements:
  • users array,
  • groups array,

writers list, A list containing the following elements:
  • users array,
  • groups array,

owners list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

Description

Set the permissions groups has on this object

Usage

reports_put_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
Arguments

id integer required. The ID of the resource that is shared.
group_ids array required. An array of one or more group IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers list, A list containing the following elements:
  • users array,
  • groups array,

writers list, A list containing the following elements:
  • users array,
  • groups array,

owners list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

--

reports_put_shares_users

Set the permissions users have on this object

Description

Set the permissions users have on this object

Usage

reports_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
Arguments

- **id** integer required. The ID of the resource that is shared.
- **user_ids** array required. An array of one or more user IDs.
- **permission_level** string required. Options are: "read", "write", or "manage".
- **share_email_body** string optional. Custom body text for e-mail sent on a share.
- **send_shared_email** boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,
- **writers** list, A list containing the following elements:
  - users array,
  - groups array,
- **owners** list, A list containing the following elements:
  - users array,
  - groups array,
- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**run_civis**  
*Evaluate an R expression in a Civis Platform container*

Description

Evaluate an R expression in a Civis Platform container

Usage

`run_civis(expr, ...)`

Arguments

- **expr** code to evaluate
- **...** arguments to `CivisFuture`
Details

`run_civis` blocks until completion. For non-blocking calls, use futures directly with `civis_platform`. Attempts are made at detecting and installing necessary packages within the container, and detecting global variables required in `expr`.

See Also

Other script utils: `civis_script()`, `fetch_output_file_ids()`, `run_template()`

Examples

```r
## Not run:
run_civis(2+2)

# specify required resources, and a specific image
run_civis(2+2,
required_resources = list(cpu = 1024, memory = 2048),
docker_image_name='image',
docker_image_tag = 'latest')

## End(Not run)
```

### Description

Run a template script

### Usage

```r
run_template(id, arguments, JSONValue = FALSE, ...)
```

### Arguments

- `id`: id of the template script.
- `arguments`: list of arguments to the script.
- `JSONValue`: bool (default FALSE) If true, returns the JSON values instead of the file_ids.
- `...`: additional arguments to `scripts_post_custom`

### Value

If JSONValue is FALSE, File ids of any run outputs are returned. If JSONValue is TRUE, JSON values of first JSON run output is returned. If there are no JSON outputs, warning message is printed and nothing is returned If there are more than 1 JSON outputs, warning message is printed and the first JSON output is returned.
saml_service_providers_delete_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

saml_service_providers_delete_shares_groups(id, group_id)

Arguments

id integer required. The ID of the resource that is shared.
group_id integer required. The ID of the group.

Value

An empty HTTP response
saml_service_providers_delete_shares_users

Revoke the permissions a user has on this object

Description
Revoke the permissions a user has on this object

Usage
saml_service_providers_delete_shares_users(id, user_id)

Arguments
id integer required. The ID of the resource that is shared.
user_id integer required. The ID of the user.

Value
An empty HTTP response

saml_service_providers_list_shares

List users and groups permissioned on this object

Description
List users and groups permissioned on this object

Usage
saml_service_providers_list_shares(id)

Arguments
id integer required. The ID of the resource that is shared.

Value
An array containing the following fields:

readers list, A list containing the following elements:
• users array,
• groups array,

writers list, A list containing the following elements:
saml_service_providers_put_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

saml_service_providers_put_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)

Arguments

id integer required. The ID of the resource that is shared.
group_ids array required. An array of one or more group IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.
Value

A list containing the following elements:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,

- **writers** list, A list containing the following elements:
  - users array,
  - groups array,

- **owners** list, A list containing the following elements:
  - users array,
  - groups array,

- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**saml_service_providers_put_shares_users**

Set the permissions users have on this object

---

Description

Set the permissions users have on this object

Usage

```python
saml_service_providers_put_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```

Arguments

- **id** integer required. The ID of the resource that is shared.
- **user_ids** array required. An array of one or more user IDs.
- **permission_level** string required. Options are: "read", "write", or "manage".
- **share_email_body** string optional. Custom body text for e-mail sent on a share.
- **send_shared_email** boolean optional. Send email to the recipients of a share.
Value

A list containing the following elements:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,

- **writers** list, A list containing the following elements:
  - users array,
  - groups array,

- **owners** list, A list containing the following elements:
  - users array,
  - groups array,

- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**scripts_delete**

Archive a script (deprecated, use archive endpoints instead)

Description

Archive a script (deprecated, use archive endpoints instead)

Usage

scripts_delete(id)

Arguments

- **id** integer required. The ID for the script.

Value

An empty HTTP response
scripts_delete_containers

Archive a container (deprecated, use archive endpoints)

Description
Archive a container (deprecated, use archive endpoints)

Usage
scripts_delete_containers(id)

Arguments
id integer required. The ID for the script.

Value
An empty HTTP response

scripts_delete_containers_projects

Remove a Container Script from a project

Description
Remove a Container Script from a project

Usage
scripts_delete_containers_projects(id, project_id)

Arguments
id integer required. The ID of the Container Script.
project_id integer required. The ID of the project.

Value
An empty HTTP response
scripts_delete_containers_runs

Cancel a run

Description
Cancel a run

Usage
scripts_delete_containers_runs(id, run_id)

Arguments
id integer required. The ID of the container.
run_id integer required. The ID of the run.

Value
An empty HTTP response

scripts_delete_containers_shares_groups

Revoke the permissions a group has on this object

Description
Revoke the permissions a group has on this object

Usage
scripts_delete_containers_shares_groups(id, group_id)

Arguments
id integer required. The ID of the resource that is shared.
group_id integer required. The ID of the group.

Value
An empty HTTP response
scripts_delete_containers_shares_users

*Revoke the permissions a user has on this object*

**Description**

Revoke the permissions a user has on this object.

**Usage**

```python
scripts_delete_containers_shares_users(id, user_id)
```

**Arguments**

- `id` integer required. The ID of the resource that is shared.
- `user_id` integer required. The ID of the user.

**Value**

An empty HTTP response

---

scripts_delete_custom

*Archive a Custom Script (deprecated, use archiving endpoints instead)*

**Description**

Archive a Custom Script (deprecated, use archiving endpoints instead).

**Usage**

```python
scripts_delete_custom(id)
```

**Arguments**

- `id` integer required.

**Value**

An empty HTTP response
scripts_delete_custom_projects

Remove a Custom Script from a project

Description

Remove a Custom Script from a project

Usage

scripts_delete_custom_projects(id, project_id)

Arguments

id integer required. The ID of the Custom Script.
project_id integer required. The ID of the project.

Value

An empty HTTP response

scripts_delete_custom_runs

Cancel a run

Description

Cancel a run

Usage

scripts_delete_custom_runs(id, run_id)

Arguments

id integer required. The ID of the custom.
run_id integer required. The ID of the run.

Value

An empty HTTP response
scripts_delete_custom_shares_groups

Revoke the permissions a group has on this object

Description
Revoke the permissions a group has on this object

Usage
scripts_delete_custom_shares_groups(id, group_id)

Arguments
id integer required. The ID of the resource that is shared.

group_id integer required. The ID of the group.

Value
An empty HTTP response

scripts_delete_custom_shares_users

Revoke the permissions a user has on this object

Description
Revoke the permissions a user has on this object

Usage
scripts_delete_custom_shares_users(id, user_id)

Arguments
id integer required. The ID of the resource that is shared.

user_id integer required. The ID of the user.

Value
An empty HTTP response
scripts_delete_javascript

Archive a JavaScript Script (deprecated, use archiving endpoints instead)

Description

Archive a JavaScript Script (deprecated, use archiving endpoints instead)

Usage

scripts_delete_javascript(id)

Arguments

id integer required.

Value

An empty HTTP response

scripts_delete_javascript_projects

Remove a JavaScript Script from a project

Description

Remove a JavaScript Script from a project

Usage

scripts_delete_javascript_projects(id, project_id)

Arguments

id integer required. The ID of the JavaScript Script.
project_id integer required. The ID of the project.

Value

An empty HTTP response
scripts_delete_javascript_runs

*Cancel a run*

**Description**

Cancel a run

**Usage**

`scripts_delete_javascript_runs(id, run_id)`

**Arguments**

- `id` integer required. The ID of the javascript.
- `run_id` integer required. The ID of the run.

**Value**

An empty HTTP response

-------

scripts_delete_javascript_shares_groups

*Revoke the permissions a group has on this object*

**Description**

Revoke the permissions a group has on this object

**Usage**

`scripts_delete_javascript_shares_groups(id, group_id)`

**Arguments**

- `id` integer required. The ID of the resource that is shared.
- `group_id` integer required. The ID of the group.

**Value**

An empty HTTP response
scripts_delete_javascript_shares_users

Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

scripts_delete_javascript_shares_users(id, user_id)

Arguments

id integer required. The ID of the resource that is shared.
user_id integer required. The ID of the user.

Value

An empty HTTP response

scripts_delete_python3

Archive a Python Script (deprecated, use archiving endpoints instead)

Description

Archive a Python Script (deprecated, use archiving endpoints instead)

Usage

scripts_delete_python3(id)

Arguments

id integer required.

Value

An empty HTTP response
scripts_delete_python3_projects

Remove a Python Script from a project

Description
Remove a Python Script from a project

Usage

scripts_delete_python3_projects(id, project_id)

Arguments

id integer required. The ID of the Python Script.
project_id integer required. The ID of the project.

Value
An empty HTTP response

scripts_delete_python3_runs

Cancel a run

Description
Cancel a run

Usage

scripts_delete_python3_runs(id, run_id)

Arguments

id integer required. The ID of the python.
run_id integer required. The ID of the run.

Value
An empty HTTP response
scripts_delete_python3_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

scripts_delete_python3_shares_groups(id, group_id)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required</td>
<td>The ID of the resource that is shared.</td>
</tr>
<tr>
<td>group_id</td>
<td>integer required</td>
<td>The ID of the group.</td>
</tr>
</tbody>
</table>

Value

An empty HTTP response

---

scripts_delete_python3_shares_users

Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

scripts_delete_python3_shares_users(id, user_id)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required</td>
<td>The ID of the resource that is shared.</td>
</tr>
<tr>
<td>user_id</td>
<td>integer required</td>
<td>The ID of the user.</td>
</tr>
</tbody>
</table>

Value

An empty HTTP response
scripts_delete_r

Archive an R Script (deprecated, use archiving endpoints instead)

Description
Archive an R Script (deprecated, use archiving endpoints instead)

Usage
scripts_delete_r(id)

Arguments
id integer required.

Value
An empty HTTP response

scripts_delete_r_projects
Remove an R Script from a project

Description
Remove an R Script from a project

Usage
scripts_delete_r_projects(id, project_id)

Arguments
id integer required. The ID of the R Script.
project_id integer required. The ID of the project.

Value
An empty HTTP response
scripts_delete_r_runs  
*Cancel a run*

**Description**

Cancel a run

**Usage**

`scripts_delete_r_runs(id, run_id)`

**Arguments**

- `id`  
  integer required. The ID of the run.
- `run_id`  
  integer required. The ID of the run.

**Value**

An empty HTTP response

---

scripts_delete_r_shares_groups

*Revoke the permissions a group has on this object*

**Description**

Revoke the permissions a group has on this object

**Usage**

`scripts_delete_r_shares_groups(id, group_id)`

**Arguments**

- `id`  
  integer required. The ID of the resource that is shared.
- `group_id`  
  integer required. The ID of the group.

**Value**

An empty HTTP response
scripts_delete_r_shares_users

Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

scripts_delete_r_shares_users(id, user_id)

Arguments

id integer required. The ID of the resource that is shared.
user_id integer required. The ID of the user.

Value

An empty HTTP response

scripts_delete_sql Archive a SQL script (deprecated, use archiving endpoints instead)

Description

Archive a SQL script (deprecated, use archiving endpoints instead)

Usage

scripts_delete_sql(id)

Arguments

id integer required.

Value

An empty HTTP response
scripts_delete_sql_projects

Remove a SQL script from a project

Description

Remove a SQL script from a project

Usage

scripts_delete_sql_projects(id, project_id)

Arguments

 id integer required. The ID of the SQL script.
 project_id integer required. The ID of the project.

Value

An empty HTTP response

scripts_delete_sql_runs

Cancel a run

Description

Cancel a run

Usage

scripts_delete_sql_runs(id, run_id)

Arguments

 id integer required. The ID of the sql.
 run_id integer required. The ID of the run.

Value

An empty HTTP response
**scripts_delete_sql_shares_groups**

*Revoke the permissions a group has on this object*

**Description**

Revoke the permissions a group has on this object.

**Usage**

`scripts_delete_sql_shares_groups(id, group_id)`

**Arguments**

- `id` : integer required. The ID of the resource that is shared.
- `group_id` : integer required. The ID of the group.

**Value**

An empty HTTP response

---

**scripts_delete_sql_shares_users**

*Revoke the permissions a user has on this object*

**Description**

Revoke the permissions a user has on this object.

**Usage**

`scripts_delete_sql_shares_users(id, user_id)`

**Arguments**

- `id` : integer required. The ID of the resource that is shared.
- `user_id` : integer required. The ID of the user.

**Value**

An empty HTTP response
scripts_get

Get details about a script

Description

Get details about a script

Usage

scripts_get(id)

Arguments

id integer required. The ID for the script.

Value

A list containing the following elements:

id integer, The ID for the script.
name string, The name of the script.
type string, The type of script.
createdAt string, The time this script was created.
updatedAt string, The time this script was last updated.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
state string, The status of the script’s last run.
finishedAt string, The time that the script’s last run finished.
category string, The category of the script.
projects array, An array containing the following fields:
  • id integer, The ID for the project.
  • name string, The name of the project.
parentId integer, The ID of the parent job that will trigger this script
userContext string, "runner" or "author", who to execute the script as when run as a template.
params array, An array containing the following fields:
  • name string, The variable’s name as used within your code.
  • label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: 'label': 'Import', 'value': 'import'

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.

publishedAsTemplateId integer, The ID of the template that this script is backing.

fromTemplateId integer, The ID of the template this script uses, if any.

templateDependentsCount integer, How many other scripts use this one as a template.

templateScriptName string, The name of the template script.

links list, A list containing the following elements:
- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

schedule list, A list containing the following elements:
- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
scripts_get

- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.

timeZone string, The time zone of this script.

lastRun list, A list containing the following elements:
  - id integer,
  - state string,
  - createdAt string, The time that the run was queued.
  - startedAt string, The time that the run started.
  - finishedAt string, The time that the run completed.
  - error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.

targetProjectId integer, Target project to which script outputs will be added.

archived string, The archival status of the requested item(s).

sql string, The raw SQL query for the script.

expandedArguments list, Expanded arguments for use in injecting into different environments.

templateScriptId integer, The ID of the template script, if any.
scripts_get_containers

View a container

Description

View a container

Usage

scripts_get_containers(id)

Arguments

id integer required. The ID for the script.

Value

A list containing the following elements:

id integer, The ID for the script.
name string, The name of the container.
type string, The type of the script (e.g Container)
createdAt string, The time this script was created.
updatedAt string, The time the script was last updated.
author list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.
state string, The status of the script's last run.
finishedAt string, The time that the script’s last run finished.
category string, The category of the script.
projects array, An array containing the following fields:
• id integer, The ID for the project.
• name string, The name of the project.
parent_id integer, The ID of the parent job that will trigger this script
userContext string, "runner" or "author", who to execute the script as when run as a template.
params array, An array containing the following fields:
• name string, The variable’s name as used within your code.
• label string, The label to present to users when asking them for the value.
scripts_get_containers

- **description string**, A short sentence or fragment describing this parameter to the end user.
- **type string**, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- **required boolean**, Whether this param is required.
- **value string**, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- **default string**, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
- **allowedValues array**, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ’Import’, ‘value’: ’import’

**arguments** list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

**isTemplate** boolean, Whether others scripts use this one as a template.

**templateDependentsCount** integer, How many other scripts use this one as a template.

**publishedAsTemplateId** integer, The ID of the template that this script is backing.

**fromTemplateId** integer, The ID of the template script.

**templateScriptName** string, The name of the template script.

**links** list, A list containing the following elements:
- **details string**, The details link to get more information about the script.
- **runs string**, The runs link to get the run information list for this script.

**schedule** list, A list containing the following elements:
- **scheduled boolean**, If the item is scheduled.
- **scheduledDays array**, Day based on numeric value starting at 0 for Sunday.
- **scheduledHours array**, Hours of the day it is scheduled on.
- **scheduledMinutes array**, Minutes of the day it is scheduled on.
- **scheduledRunsPerHour integer**, Alternative to scheduled minutes, number of times to run per hour.

**notifications** list, A list containing the following elements:
- **urls array**, URLs to receive a POST request at job completion
- **successEmailSubject string**, Custom subject line for success e-mail.
- **successEmailBody string**, Custom body text for success e-mail, written in Markdown.
- **successEmailAddresses array**, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

requiredResources list, A list containing the following elements:
• cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares.
• memory integer, The amount of RAM to allocate for the container (in MB).
• diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

repoHttpUri string, The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.
repoRef string, The tag or branch of the github repo to clone into the container.
remoteHostCredentialId integer, The id of the database credentials to pass into the environment of the container.

gitCredentialId integer, The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you’ve submitted will be used. Unnecessary if no git repo is specified or the git repo is public.

dockerCommand string, The command to run on the container. Will be run via sh as: ["sh", ",-c", dockerCommand]. Defaults to the Docker image’s ENTRYPOINT/CMD.
dockerImageName string, The name of the docker image to pull from DockerHub.
dockerImageTag string, The tag of the docker image to pull from DockerHub.
instanceType string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.
cancelTimeout integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.
lastRun

list, A list containing the following elements:

• id integer,
• state string,
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.
• error string, The error message for this run, if present.

timeZone

string, The time zone of this script.

hidden

boolean, The hidden status of the item.

archived

string, The archival status of the requested item(s).

targetProjectId

integer, Target project to which script outputs will be added.

---

scripts_get_containers_runs

*Check status of a run*

**Description**

Check status of a run

**Usage**

`scripts_get_containers_runs(id, run_id)`

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of the container.</td>
</tr>
<tr>
<td>run_id</td>
<td>integer required. The ID of the run.</td>
</tr>
</tbody>
</table>

**Value**

A list containing the following elements:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer, The ID of the run.</td>
</tr>
<tr>
<td>containerId</td>
<td>integer, The ID of the container.</td>
</tr>
<tr>
<td>state</td>
<td>string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.</td>
</tr>
<tr>
<td>isCancelRequested</td>
<td>boolean, True if run cancel requested, else false.</td>
</tr>
<tr>
<td>startedAt</td>
<td>string, The time the last run started at.</td>
</tr>
<tr>
<td>finishedAt</td>
<td>string, The time the last run completed.</td>
</tr>
<tr>
<td>error</td>
<td>string, The error, if any, returned by the run.</td>
</tr>
</tbody>
</table>
scripts_get_custom  

Get a Custom Script

Description
Get a Custom Script

Usage
scripts_get_custom(id)

Arguments

id  integer required.

Value
A list containing the following elements:

id  integer, The ID for the script.
name  string, The name of the script.
type  string, The type of the script (e.g. Custom)
createdAt  string, The time this script was created.
updatedAt  string, The time the script was last updated.
author  list, A list containing the following elements:
• id  integer, The ID of this user.
• name  string, This user’s name.
• username  string, This user’s username.
• initials  string, This user’s initials.
• online  boolean, Whether this user is online.
state  string, The status of the script’s last run.
finishedAt  string, The time that the script’s last run finished.
category  string,
projects  array, An array containing the following fields:
• id  integer, The ID for the project.
• name  string, The name of the project.
parentid  integer, The ID of the parent job that will trigger this script
params  array, An array containing the following fields:
• name  string, The variable’s name as used within your code.
• label  string, The label to present to users when asking them for the value.
• description  string, A short sentence or fragment describing this parameter to the end user.
• type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
• required boolean, Whether this param is required.
• value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
• default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
• allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: 'Import', 'value': 'import''

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.
publishedAsTemplateId integer, The ID of the template that this script is backing.

fromTemplateId integer, The ID of the template script.

uiReportUrl integer, The url of the custom HTML.

uiReportId integer, The id of the report with the custom HTML.

uiReportProvideAPIKey boolean, Whether the ui report requests an API Key from the report viewer.

templateScriptName string, The name of the template script.
templateNote string, The template’s note.
remoteHostId integer, The remote host ID that this script will connect to.
credentialId integer, The credential that this script will use.

codePreview string, The code that this script will run with arguments inserted.
schedule list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
scripts_get_custom

- **successEmailFromName** string, Name from which success emails are sent; defaults to "Civis."
- **successEmailReplyTo** string, Address for replies to success emails; defaults to the author of the job.
- **failureEmailAddresses** array, Addresses to notify by e-mail when the job fails.
- **stallWarningMinutes** integer, Stall warning emails will be sent after this amount of minutes.
- **successOn** boolean, If success email notifications are on.
- **failureOn** boolean, If failure email notifications are on.

**runningAs** list, A list containing the following elements:
- **id** integer, The ID of this user.
- **name** string, This user’s name.
- **username** string, This user’s username.
- **initials** string, This user’s initials.
- **online** boolean, Whether this user is online.

**timeZone** string, The time zone of this script.

**lastRun** list, A list containing the following elements:
- **id** integer,
- **state** string,
- **createdAt** string, The time that the run was queued.
- **startedAt** string, The time that the run started.
- **finishedAt** string, The time that the run completed.
- **error** string, The error message for this run, if present.

**hidden** boolean, The hidden status of the item.

**archived** string, The archival status of the requested item(s).

**targetProjectId** integer, Target project to which script outputs will be added.

**lastSuccessfulRun** list, A list containing the following elements:
- **id** integer,
- **state** string,
- **createdAt** string, The time that the run was queued.
- **startedAt** string, The time that the run started.
- **finishedAt** string, The time that the run completed.
- **error** string, The error message for this run, if present.
scripts_get_custom_runs

*Check status of a run*

**Description**

Check status of a run

**Usage**

```
scripts_get_custom_runs(id, run_id)
```

**Arguments**

- `id`  
  integer required. The ID of the custom.

- `run_id`  
  integer required. The ID of the run.

**Value**

A list containing the following elements:

- `id`  
  integer, The ID of the run.

- `customId`  
  integer, The ID of the custom.

- `state`  
  string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

- `isCancelRequested`  
  boolean, True if run cancel requested, else false.

- `startedAt`  
  string, The time the last run started at.

- `finishedAt`  
  string, The time the last run completed.

- `error`  
  string, The error, if any, returned by the run.

---

scripts_get_javascript

*Get a JavaScript Script*

**Description**

Get a JavaScript Script

**Usage**

```
scripts_get_javascript(id)
```
Arguments

id  integer required.

Value

A list containing the following elements:

id  integer, The ID for the script.
name  string, The name of the script.
type  string, The type of the script (e.g SQL, Container, Python, R, JavaScript)
createdAt  string, The time this script was created.
updatedAt  string, The time the script was last updated.
author  list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
state  string, The status of the script’s last run.
finishedAt  string, The time that the script’s last run finished.
category  string, The category of the script.
projects  array, An array containing the following fields:
  • id integer, The ID for the project.
  • name string, The name of the project.
parentId  integer, The ID of the parent job that will trigger this script
userContext  string, "runner" or "author", who to execute the script as when run as a template.
params  array, An array containing the following fields:
  • name string, The variable’s name as used within your code.
  • label string, The label to present to users when asking them for the value.
  • description string, A short sentence or fragment describing this parameter to the end user.
  • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  • required boolean, Whether this param is required.
  • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
• allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.

publishedAsTemplateId integer, The ID of the template that this script is backing.

fromTemplateId integer, The ID of the template this script uses, if any.

templateDependentsCount integer, How many other scripts use this one as a template.

templateScriptName string, The name of the template script.

links list, A list containing the following elements:
• details string, The details link to get more information about the script.
• runs string, The runs link to get the run information list for this script.

schedule list, A list containing the following elements:
• scheduled boolean, If the item is scheduled.
• scheduledDays array, Day based on numeric value starting at 0 for Sunday.
• scheduledHours array, Hours of the day it is scheduled on.
• scheduledMinutes array, Minutes of the day it is scheduled on.
• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
• urls array, URLs to receive a POST request at job completion
• successEmailSubject string, Custom subject line for success e-mail.
• successEmailBody string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
scripts_get_javascript_git_commits

- name string, This user’s name.
- username string, This user’s username.
- initials string, This user’s initials.
- online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.
timeZone string, The time zone of this script.
lastRun list, A list containing the following elements:
  - id integer,
  - state string,
  - createdAt string, The time that the run was queued.
  - startedAt string, The time that the run started.
  - finishedAt string, The time that the run completed.
  - error string, The error message for this run, if present.
hidden boolean, The hidden status of the item.
targetProjectId integer, Target project to which script outputs will be added.
archived string, The archival status of the requested item(s).
source string, The body/text of the script.
remoteHostId integer, The remote host ID that this script will connect to.
credentialId integer, The credential that this script will use.

---

scripts_get_javascript_git_commits

*Get file contents at commit_hash*

**Description**

Get file contents at commit_hash

**Usage**

`scripts_get_javascript_git_commits(id, commit_hash)`

**Arguments**

- **id** integer required. The ID of the file.
- **commit_hash** string required. The SHA (full or shortened) of the desired git commit.

**Value**

A list containing the following elements:

- **content** string, The file’s contents.
- **type** string, The file’s type.
- **size** integer, The file’s size.
- **fileHash** string, The SHA of the file.
scripts_get_javascript_runs

*Check status of a run*

**Description**

Check status of a run

**Usage**

`scripts_get_javascript_runs(id, run_id)`

**Arguments**

- **id** integer required. The ID of the javascript.
- **run_id** integer required. The ID of the run.

**Value**

A list containing the following elements:

- **id** integer, The ID of the run.
- **javascriptId** integer, The ID of the javascript.
- **state** string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
- **isCancelRequested** boolean, True if run cancel requested, else false.
- **startedAt** string, The time the last run started at.
- **finishedAt** string, The time the last run completed.
- **error** string, The error, if any, returned by the run.

---

scripts_get_python3  

*Get a Python Script*

**Description**

Get a Python Script

**Usage**

`scripts_get_python3(id)`

**Arguments**

- **id** integer required.
Value

A list containing the following elements:

- id: integer, The ID for the script.
- name: string, The name of the script.
- type: string, The type of the script (e.g. SQL, Container, Python, R, JavaScript)
- createdAt: string, The time this script was created.
- updatedAt: string, The time the script was last updated.
- author: list, A list containing the following elements:
  - id: integer, The ID of this user.
  - name: string, This user’s name.
  - username: string, This user’s username.
  - initials: string, This user’s initials.
  - online: boolean, Whether this user is online.
- state: string, The status of the script’s last run.
- finishedAt: string, The time that the script’s last run finished.
- category: string, The category of the script.
- projects: array, An array containing the following fields:
  - id: integer, The ID for the project.
  - name: string, The name of the project.
- parentId: integer, The ID of the parent job that will trigger this script
- userContext: string, "runner" or "author", who to execute the script as when run as a template.
- params: array, An array containing the following fields:
  - name: string, The variable’s name as used within your code.
  - label: string, The label to present to users when asking them for the value.
  - description: string, A short sentence or fragment describing this parameter to the end user.
  - type: string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  - required: boolean, Whether this param is required.
  - value: string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  - default: string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  - allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ’Import’, ‘value’: ’import’
- arguments: list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
isTemplate boolean, Whether others scripts use this one as a template.

publishedAsTemplateId integer, The ID of the template that this script is backing.

fromTemplateId integer, The ID of the template this script uses, if any.

templateDependentsCount integer, How many other scripts use this one as a template.

templateScriptName string, The name of the template script.

links list, A list containing the following elements:
  • details string, The details link to get more information about the script.
  • runs string, The runs link to get the run information list for this script.

schedule list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  • successOn boolean, If success email notifications are on.
  • failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.
**scripts_get_python3_git_commits**

*Get file contents at commit_hash*

---

**Description**

Get file contents at commit_hash

**Usage**

```
scripts_get_python3_git_commits(id, commit_hash)
```
**scripts_get_python3_runs**

**Arguments**

- **id** integer required. The ID of the file.
- **commit_hash** string required. The SHA (full or shortened) of the desired git commit.

**Value**

A list containing the following elements:

- **content** string, The file’s contents.
- **type** string, The file’s type.
- **size** integer, The file’s size.
- **fileHash** string, The SHA of the file.

---

**scripts_get_python3_runs**

*Check status of a run*

**Description**

Check status of a run

**Usage**

`scripts_get_python3_runs(id, run_id)`

**Arguments**

- **id** integer required. The ID of the python.
- **run_id** integer required. The ID of the run.

**Value**

A list containing the following elements:

- **id** integer, The ID of the run.
- **pythonId** integer, The ID of the python.
- **state** string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
- **isCancelRequested** boolean, True if run cancel requested, else false.
- **startedAt** string, The time the last run started at.
- **finishedAt** string, The time the last run completed.
- **error** string, The error, if any, returned by the run.
Description

Get an R Script

Usage

scripts_get_r(id)

Arguments

id integer required.

Value

A list containing the following elements:

id integer, The ID for the script.
name string, The name of the script.
type string, The type of the script (e.g SQL, Container, Python, R, JavaScript)
createdAt string, The time this script was created.
updatedAt string, The time the script was last updated.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user's name.
  • username string, This user's username.
  • initials string, This user's initials.
  • online boolean, Whether this user is online.
state string, The status of the script's last run.
finishedAt string, The time that the script's last run finished.
category string, The category of the script.
projects array, An array containing the following fields:
  • id integer, The ID for the project.
  • name string, The name of the project.
parentId integer, The ID of the parent job that will trigger this script
userContext string, "runner" or "author", who to execute the script as when run as a template.
params array, An array containing the following fields:
  • name string, The variable's name as used within your code.
  • label string, The label to present to users when asking them for the value.
* description string, A short sentence or fragment describing this parameter to the end user.
* type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
* required boolean, Whether this param is required.
* value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
* default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
* allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: 'Import', 'value': 'import'

**arguments** list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

**isTemplate** boolean, Whether others scripts use this one as a template.

**publishedAsTemplateId** integer, The ID of the template that this script is backing.

**fromTemplateId** integer, The ID of the template this script uses, if any.

**templateDependentsCount** integer, How many other scripts use this one as a template.

**templateScriptName** string, The name of the template script.

**links** list, A list containing the following elements:

- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

**schedule** list, A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

**notifications** list, A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.

timeZone string, The time zone of this script.

eRunAt list, A list containing the following elements:
• id integer, 
• state string, 
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.
• error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.

targetProjectId integer, Target project to which script outputs will be added.
archived string, The archival status of the requested item(s).

requiredResources list, A list containing the following elements:
• cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
• memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
• diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

instanceType string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

source string, The body/text of the script.
cancelTimeout integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

dockerImageTag string, The tag of the docker image to pull from DockerHub.

---

**scripts_get_r_git_commits**

*Get file contents at commit_hash*

**Description**

Get file contents at commit_hash

**Usage**

```
scripts_get_r_git_commits(id, commit_hash)
```

**Arguments**

- **id** integer required. The ID of the file.
- **commit_hash** string required. The SHA (full or shortened) of the desired git commit.

**Value**

A list containing the following elements:

- **content** string, The file’s contents.
- **type** string, The file’s type.
- **size** integer, The file’s size.
- **fileHash** string, The SHA of the file.

---

**scripts_get_r_runs**

*Check status of a run*

**Description**

Check status of a run

**Usage**

```
scripts_get_r_runs(id, run_id)
```

**Arguments**

- **id** integer required. The ID of the r.
- **run_id** integer required. The ID of the run.
Value

A list containing the following elements:

- **id**: integer, The ID of the run.
- **rId**: integer, The ID of the r.
- **state**: string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
- **isCancelRequested**: boolean, True if run cancel requested, else false.
- **startedAt**: string, The time the last run started at.
- **finishedAt**: string, The time the last run completed.
- **error**: string, The error, if any, returned by the run.

---

**scripts_get_sql**

*Get a SQL script*

Description

Get a SQL script

Usage

```python
scripts_get_sql(id)
```

Arguments

- **id**: integer required.

Value

A list containing the following elements:

- **id**: integer, The ID for the script.
- **name**: string, The name of the script.
- **type**: string, The type of the script (e.g SQL, Container, Python, R, JavaScript)
- **createdAt**: string, The time this script was created.
- **updatedAt**: string, The time the script was last updated.
- **author**: list, A list containing the following elements:
  - **id**: integer, The ID of this user.
  - **name**: string, This user’s name.
  - **username**: string, This user’s username.
  - **initials**: string, This user’s initials.
  - **online**: boolean, Whether this user is online.
state string, The status of the script’s last run.
finishedAt string, The time that the script’s last run finished.
category string, The category of the script.
projects array, An array containing the following fields:
  • id integer, The ID for the project.
  • name string, The name of the project.
parentId integer, The ID of the parent job that will trigger this script.
userContext string, "runner" or "author", who to execute the script as when run as a template.
params array, An array containing the following fields:
  • name string, The variable’s name as used within your code.
  • label string, The label to present to users when asking them for the value.
  • description string, A short sentence or fragment describing this parameter to the end user.
  • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  • required boolean, Whether this param is required.
  • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: 'Import', 'value': 'import'’
arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
isTemplate boolean, Whether others scripts use this one as a template.
publishedAsTemplateId integer, The ID of the template that this script is backing.
fromTemplateId integer, The ID of the template this script uses, if any.
templateDependentsCount integer, How many other scripts use this one as a template.
templateScriptName string, The name of the template script.
links list, A list containing the following elements:
  • details string, The details link to get more information about the script.
  • runs string, The runs link to get the run information list for this script.
schedule list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
• scheduledHours array, Hours of the day it is scheduled on.
• scheduledMinutes array, Minutes of the day it is scheduled on.
• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
• urls array, URLs to receive a POST request at job completion
• successEmailSubject string, Custom subject line for success e-mail.
• successEmailBody string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.
timeZone string, The time zone of this script.

lastRun list, A list containing the following elements:
• id integer,
• state string,
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.
• error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.
targetProjectId integer, Target project to which script outputs will be added.

archived string, The archival status of the requested item(s).
sql string, The raw SQL query for the script.
scripts_get_sql_git_commits

Get file contents at commit_hash

Description
Get file contents at commit_hash

Usage
scripts_get_sql_git_commits(id, commit_hash)

Arguments
id integer required. The ID of the file.
commit_hash string required. The SHA (full or shortened) of the desired git commit.

Value
A list containing the following elements:
content string, The file’s contents.
type string, The file’s type.
size integer, The file’s size.
fileHash string, The SHA of the file.
scripts_get_sql_runs  

Description

Check status of a run

Usage

scripts_get_sql_runs(id, run_id)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of the sql.</td>
</tr>
<tr>
<td>run_id</td>
<td>integer required. The ID of the run.</td>
</tr>
</tbody>
</table>

Value

A list containing the following elements:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer, The ID of this run.</td>
</tr>
<tr>
<td>sqlId</td>
<td>integer, The ID of this sql.</td>
</tr>
<tr>
<td>state</td>
<td>string, The state of this run.</td>
</tr>
<tr>
<td>isCancelRequested</td>
<td>boolean, True if run cancel requested, else false.</td>
</tr>
<tr>
<td>startedAt</td>
<td>string, The time the last run started.</td>
</tr>
<tr>
<td>finishedAt</td>
<td>string, The time that this run finished.</td>
</tr>
<tr>
<td>error</td>
<td>string, The error message for this run, if present.</td>
</tr>
<tr>
<td>output</td>
<td>array, An array containing the following fields:</td>
</tr>
<tr>
<td></td>
<td>• outputName string, The name of the output file.</td>
</tr>
<tr>
<td></td>
<td>• fileId integer, The unique ID of the output file.</td>
</tr>
<tr>
<td></td>
<td>• path string, The temporary link to download this output file, valid for 36 hours.</td>
</tr>
</tbody>
</table>
scripts_list

List Scripts

Description
List Scripts

Usage

scripts_list(
    type = NULL,
    category = NULL,
    author = NULL,
    status = NULL,
    hidden = NULL,
    archived = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)

Arguments

type            string optional. If specified, return items of these types. The valid types are sql, python3, javascript, r, and containers.
category        string optional. A job category for filtering scripts. Must be one of script, import, export, and enhancement.
author          string optional. If specified, return items from this author. Must use user IDs. A comma separated list of IDs is also accepted to return items from multiple authors.
status          string optional. If specified, returns items with one of these statuses. It accepts a comma-separated list, possible values are 'running', 'failed', 'succeeded', 'idle', 'scheduled'.
hidden          boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
arched          string optional. The archival status of the requested item(s).
limit           integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
page_num        integer optional. Page number of the results to return. Defaults to the first page, 1.
order           string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at, last_run.updated_at.
order_dir       string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.
### Value

An array containing the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer, The ID for the script.</td>
</tr>
<tr>
<td>name</td>
<td>string, The name of the script.</td>
</tr>
<tr>
<td>type</td>
<td>string, The type of the script (e.g SQL, Container, Python, R, JavaScript)</td>
</tr>
<tr>
<td>createdAt</td>
<td>string, The time this script was created.</td>
</tr>
<tr>
<td>updatedAt</td>
<td>string, The time the script was last updated.</td>
</tr>
<tr>
<td>author</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• id integer, The ID of this user.</td>
</tr>
<tr>
<td></td>
<td>• name string, This user’s name.</td>
</tr>
<tr>
<td></td>
<td>• username string, This user’s username.</td>
</tr>
<tr>
<td></td>
<td>• initials string, This user’s initials.</td>
</tr>
<tr>
<td></td>
<td>• online boolean, Whether this user is online.</td>
</tr>
<tr>
<td>state</td>
<td>string, The status of the script's last run.</td>
</tr>
<tr>
<td>finishedAt</td>
<td>string, The time that the script's last run finished.</td>
</tr>
<tr>
<td>projects</td>
<td>array, An array containing the following fields:</td>
</tr>
<tr>
<td></td>
<td>• id integer, The ID for the project.</td>
</tr>
<tr>
<td></td>
<td>• name string, The name of the project.</td>
</tr>
<tr>
<td>parentId</td>
<td>integer, The ID of the parent job that will trigger this script</td>
</tr>
<tr>
<td>isTemplate</td>
<td>boolean, Whether others scripts use this one as a template.</td>
</tr>
<tr>
<td>fromTemplateId</td>
<td>integer, The ID of the template this script uses, if any.</td>
</tr>
<tr>
<td>links</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• details string, The details link to get more information about the script.</td>
</tr>
<tr>
<td></td>
<td>• runs string, The runs link to get the run information list for this script.</td>
</tr>
<tr>
<td>timeZone</td>
<td>string, The time zone of this script.</td>
</tr>
<tr>
<td>lastRun</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• id integer,</td>
</tr>
<tr>
<td></td>
<td>• state string,</td>
</tr>
<tr>
<td></td>
<td>• createdAt string, The time that the run was queued.</td>
</tr>
<tr>
<td></td>
<td>• startedAt string, The time that the run started.</td>
</tr>
<tr>
<td></td>
<td>• finishedAt string, The time that the run completed.</td>
</tr>
<tr>
<td></td>
<td>• error string, The error message for this run, if present.</td>
</tr>
<tr>
<td>archived</td>
<td>string, The archival status of the requested item(s).</td>
</tr>
<tr>
<td>templateScriptId</td>
<td>integer, The ID of the template script, if any.</td>
</tr>
</tbody>
</table>
scripts_list_containers_projects

List the projects a Container Script belongs to

Description

List the projects a Container Script belongs to

Usage

scripts_list_containers_projects(id, hidden = NULL)

Arguments

id
integer required. The ID of the Container Script.

hidden
boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id
integer, The ID for this project.

author
list, A list containing the following elements:

• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

name
string, The name of this project.

description
string, A description of the project.

users
array, An array containing the following fields:

• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

autoShare
boolean,

createdAt
string,

updatedAt
string,

archived
string, The archival status of the requested item(s).
scripts_list_containers_runs

List runs for the given container

Description

List runs for the given container

Usage

scripts_list_containers_runs(
    id,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)

Arguments

id              integer required. The ID of the container.
limit           integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num        integer optional. Page number of the results to return. Defaults to the first page, 1.
order           string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir       string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id              integer, The ID of the run.
containerId     integer, The ID of the container.
state           string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested boolean, True if run cancel requested, else false.
startedAt       string, The time the last run started at.
finishedAt      string, The time the last run completed.
error           string, The error, if any, returned by the run.
scripts_list_containers_runs_logs

*Get the logs for a run*

**Description**

Get the logs for a run

**Usage**

```r
scripts_list_containers_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

**Arguments**

- **id**
  - integer required. The ID of the container.
- **run_id**
  - integer required. The ID of the run.
- **last_id**
  - integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted. Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
- **limit**
  - integer optional. The maximum number of log messages to return. Default of 10000.

**Value**

An array containing the following fields:

- **id**
  - integer. The ID of the log.
- **createdAt**
  - string. The time the log was created.
- **message**
  - string. The log message.
- **level**
  - string. The level of the log. One of unknown,fatal,error,warn,info,debug.

---

scripts_list_containers_runs_outputs

*List the outputs for a run*

**Description**

List the outputs for a run
Usage

```python
scripts_list_containers_runs_outputs(
    id,
    run_id,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```

Arguments

- **id**: integer required. The ID of the container script.
- **run_id**: integer required. The ID of the run.
- **limit**: integer optional. Number of results to return. Defaults to its maximum of 50.
- **page_num**: integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order**: string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
- **order_dir**: string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

- **objectType**: string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
- **objectId**: integer, The ID of the output.
- **name**: string, The name of the output.
- **link**: string, The hypermedia link to the output.
- **value**: string,

---

**scripts_list_containers_shares**

*List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```python
scripts_list_containers_shares(id)
```
**Argument**

id
text. Required. The ID of the resource that is shared.

**Value**

An array containing the following fields:

- **readers**
  - list, A list containing the following elements:
    - users array,
    - groups array,

- **writers**
  - list, A list containing the following elements:
    - users array,
    - groups array,

- **owners**
  - list, A list containing the following elements:
    - users array,
    - groups array,

- **totalUserShares**
  - integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

- **totalGroupShares**
  - integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**scripts_list_custom**  
*List Custom Scripts*

**Description**

List Custom Scripts

**Usage**

```python
scripts_list_custom(
    from_template_id = NULL,
    author = NULL,
    status = NULL,
    hidden = NULL,
    archived = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)```
Arguments

- `from_template_id` (string, optional): If specified, return scripts based on the template with this ID. Specify multiple IDs as a comma-separated list.
- `author` (string, optional): If specified, return items from this author. Must use user IDs. A comma-separated list of IDs is also accepted to return items from multiple authors.
- `status` (string, optional): If specified, returns items with one of these statuses. It accepts a comma-separated list, possible values are 'running', 'failed', 'succeeded', 'idle', 'scheduled'.
- `hidden` (boolean, optional): If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
- `archived` (string, optional): The archival status of the requested item(s).
- `limit` (integer, optional): Number of results to return. Defaults to its maximum of 50.
- `page_num` (integer, optional): Page number of the results to return. Defaults to the first page, 1.
- `order` (string, optional): The field on which to order the result set. Defaults to `updated_at`. Must be one of: `updated_at`, `name`, `created_at`.
- `order_dir` (string, optional): Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

- `id` (integer): The ID for the script.
- `name` (string): The name of the script.
- `type` (string): The type of the script (e.g. Custom).
- `createdAt` (string): The time this script was created.
- `updatedAt` (string): The time the script was last updated.
- `author` (list): A list containing the following elements:
  - `id` (integer): The ID of this user.
  - `name` (string): This user’s name.
  - `username` (string): This user’s username.
  - `initials` (string): This user’s initials.
  - `online` (boolean): Whether this user is online.
- `state` (string): The status of the script’s last run.
- `finishedAt` (string): The time that the script’s last run finished.
- `parentId` (integer): The ID of the parent job that will trigger this script

...
**scripts_list_custom_projects**

*List the projects a Custom Script belongs to*

**Description**

List the projects a Custom Script belongs to

**Usage**

```r
scripts_list_custom_projects(id, hidden = NULL)
```

**Arguments**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of the Custom Script.</td>
</tr>
<tr>
<td>hidden</td>
<td>boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.</td>
</tr>
</tbody>
</table>
Value

An array containing the following fields:

- id: integer, The ID for this project.
- author: list, A list containing the following elements:
  - id: integer, The ID of this user.
  - name: string, This user’s name.
  - username: string, This user’s username.
  - initials: string, This user’s initials.
  - online: boolean, Whether this user is online.
- name: string, The name of this project.
- description: string, A description of the project.
- users: array, An array containing the following fields:
  - id: integer, The ID of this user.
  - name: string, This user’s name.
  - username: string, This user’s username.
  - initials: string, This user’s initials.
  - online: boolean, Whether this user is online.
- autoShare: boolean,
- createdAt: string,
- updatedAt: string,
- archived: string, The archival status of the requested item(s).

---

scripts_list_custom_runs

*List runs for the given custom*

---

Description

List runs for the given custom

Usage

```python
scripts_list_custom_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```
scripts_list_custom_runs_logs

Get the logs for a run

Description
Get the logs for a run

Usage

scripts_list_custom_runs_logs(id, run_id, last_id = NULL, limit = NULL)

Arguments

id integer required. The ID of the custom.
run_id integer required. The ID of the run.
last_id integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted. Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit integer optional. The maximum number of log messages to return. Default of 10000.
scripts_list_custom_runs_outputs

List the outputs for a run

Description

List the outputs for a run

Usage

scripts_list_custom_runs_outputs(
  id,
  run_id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)

Arguments

id integer required. The ID of the custom script.
run_id integer required. The ID of the run.
limit integer optional. Number of results to return. Defaults to its maximum of 50.
page_num integer optional. Page number of the results to return. Defaults to the first page, 1.
order string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
order_dir string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.
scripts_list_custom_shares

List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

scripts_list_custom_shares(id)

Arguments

id integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers list, A list containing the following elements:
  • users array,
  • groups array,

writers list, A list containing the following elements:
  • users array,
  • groups array,

owners list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.
**scripts_list_history**  
*Get the run history and outputs of this script*

**Description**
Get the run history and outputs of this script

**Usage**
```
scripts_list_history(id)
```

**Arguments**
- **id**  
  integer required. The ID for the script.

**Value**
A list containing the following elements:
- **id**  
  integer, The ID of this run.
- **sqlId**  
  integer, The ID of this sql.
- **state**  
  string, The state of this run.
- **isCancelRequested**  
  boolean, True if run cancel requested, else false.
- **finishedAt**  
  string, The time that this run finished.
- **error**  
  string, The error message for this run, if present.
- **output**  
  array, An array containing the following fields:
  - **outputName**  
    string, The name of the output file.
  - **fileId**  
    integer, The unique ID of the output file.
  - **path**  
    string, The temporary link to download this output file, valid for 36 hours.

**scripts_list_javascript_git**  
*Get the git metadata attached to an item*

**Description**
Get the git metadata attached to an item

**Usage**
```
scripts_list_javascript_git(id)
```
Arguments

id  integer required. The ID of the file.

Value

A list containing the following elements:

gitRef  string, A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.

gitBranch  string, The git branch that the file is on.

gitPath  string, The path of the file in the repository.

gitRepo  list, A list containing the following elements:
  • id integer, The ID for this git repository.
  • repoUrl string, The URL for this git repository.
  • createdAt string,
  • updatedAt string,

pullFromGit  boolean, Automatically pull latest commit from git. Only works for scripts.

---

counts_list_javascript_git_commits

Get the git commits for an item

Description

Get the git commits for an item

Usage

scripts_list_javascript_git_commits(id)

Arguments

id  integer required. The ID of the file.

Value

A list containing the following elements:

commitHash  string, The SHA of the commit.

authorName  string, The name of the commit’s author.

date  string, The commit’s timestamp.

message  string, The commit message.
scripts_list_javascript_projects

List the projects a JavaScript Script belongs to

Description

List the projects a JavaScript Script belongs to

Usage

scripts_list_javascript_projects(id, hidden = NULL)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of the JavaScript Script.</td>
</tr>
<tr>
<td>hidden</td>
<td>boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.</td>
</tr>
</tbody>
</table>

Value

An array containing the following fields:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer, The ID for this project.</td>
</tr>
<tr>
<td>author</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• id integer, The ID of this user.</td>
</tr>
<tr>
<td></td>
<td>• name string, This user’s name.</td>
</tr>
<tr>
<td></td>
<td>• username string, This user’s username.</td>
</tr>
<tr>
<td></td>
<td>• initials string, This user’s initials.</td>
</tr>
<tr>
<td></td>
<td>• online boolean, Whether this user is online.</td>
</tr>
<tr>
<td>name</td>
<td>string, The name of this project.</td>
</tr>
<tr>
<td>description</td>
<td>string, A description of the project.</td>
</tr>
<tr>
<td>users</td>
<td>array, An array containing the following fields:</td>
</tr>
<tr>
<td></td>
<td>• id integer, The ID of this user.</td>
</tr>
<tr>
<td></td>
<td>• name string, This user’s name.</td>
</tr>
<tr>
<td></td>
<td>• username string, This user’s username.</td>
</tr>
<tr>
<td></td>
<td>• initials string, This user’s initials.</td>
</tr>
<tr>
<td></td>
<td>• online boolean, Whether this user is online.</td>
</tr>
<tr>
<td>autoShare</td>
<td>boolean,</td>
</tr>
<tr>
<td>createdAt</td>
<td>string,</td>
</tr>
<tr>
<td>updatedAt</td>
<td>string,</td>
</tr>
<tr>
<td>archived</td>
<td>string, The archival status of the requested item(s).</td>
</tr>
</tbody>
</table>
scripts_list_javascript_runs

*List runs for the given javascript*

**Description**

List runs for the given javascript

**Usage**

```
scripts_list_javascript_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

**Arguments**

- **id** integer required. The ID of the javascript.
- **limit** integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
- **page_num** integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order** string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
- **order_dir** string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

**Value**

An array containing the following fields:

- **id** integer, The ID of the run.
- **javascriptId** integer, The ID of the javascript.
- **state** string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
- **isCancelRequested** boolean, True if run cancel requested, else false.
- **startedAt** string, The time the last run started at.
- **finishedAt** string, The time the last run completed.
- **error** string, The error, if any, returned by the run.
scripts_list_javascript_runs_logs

Get the logs for a run

Description

Get the logs for a run

Usage

scripts_list_javascript_runs_logs(id, run_id, last_id = NULL, limit = NULL)

Arguments

id  integer required. The ID of the javascript.
run_id  integer required. The ID of the run.
last_id  integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted. Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
limit  integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

id  integer, The ID of the log.
createdAt  string, The time the log was created.
message  string, The log message.
level  string, The level of the log. One of unknown,fatal,error,warn,info,debug.

scripts_list_javascript_runs_outputs

List the outputs for a run

Description

List the outputs for a run
Usage

```python
scripts_list_javascript_runs_outputs(
    id,
    run_id,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```

Arguments

- **id**: integer required. The ID of the javascript script.
- **run_id**: integer required. The ID of the run.
- **limit**: integer optional. Number of results to return. Defaults to its maximum of 50.
- **page_num**: integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order**: string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
- **order_dir**: string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

- **objectType**: string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
- **objectId**: integer, The ID of the output.
- **name**: string, The name of the output.
- **link**: string, The hypermedia link to the output.
- **value**: string,

---

```python
scripts_list_javascript_shares

List users and groups permissioned on this object
```

Description

List users and groups permissioned on this object

Usage

```python
scripts_list_javascript_shares(id)
```
Arguments

id  integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers  list, A list containing the following elements:
  • users array,
  • groups array,

writers  list, A list containing the following elements:
  • users array,
  • groups array,

owners  list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares  integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares  integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

Get the git metadata attached to an item

Get the git metadata attached to an item

scripts_list_python3_git(id)
Value

A list containing the following elements:

- **gitRef** string, A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
- **gitBranch** string, The git branch that the file is on.
- **gitPath** string, The path of the file in the repository.
- **gitRepo** list, A list containing the following elements:
  - id integer, The ID for this git repository.
  - repoUrl string, The URL for this git repository.
  - createdAt string,
  - updatedAt string,
- **pullFromGit** boolean, Automatically pull latest commit from git. Only works for scripts.

---

**scripts_list_python3_git_commits**

*Get the git commits for an item*

Description

Get the git commits for an item

Usage

**scripts_list_python3_git_commits(id)**

Arguments

- id integer required. The ID of the file.

Value

A list containing the following elements:

- **commitHash** string, The SHA of the commit.
- **authorName** string, The name of the commit’s author.
- **date** string, The commit’s timestamp.
- **message** string, The commit message.
scripts_list_python3_projects

*List the projects a Python Script belongs to*

**Description**

List the projects a Python Script belongs to.

**Usage**

```python
scripts_list_python3_projects(id, hidden = NULL)
```

**Arguments**

- **id** integer required. The ID of the Python Script.
- **hidden** boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

**Value**

An array containing the following fields:

- **id** integer, The ID for this project.
- **author** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **name** string, The name of this project.
- **description** string, A description of the project.
- **users** array, An array containing the following fields:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **autoShare** boolean,
- **createdAt** string,
- **updatedAt** string,
- **archived** string, The archival status of the requested item(s).
scripts_list_python3_runs

*List runs for the given python*

**Description**

List runs for the given python

**Usage**

```python
scripts_list_python3_runs(
    id,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```

**Arguments**

- **id**
  - integer required. The ID of the python.

- **limit**
  - integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.

- **page_num**
  - integer optional. Page number of the results to return. Defaults to the first page, 1.

- **order**
  - string optional. The field on which to order the result set. Defaults to id. Must be one of: id.

- **order_dir**
  - string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

**Value**

An array containing the following fields:

- **id**
  - integer, The ID of the run.

- **pythonId**
  - integer, The ID of the python.

- **state**
  - string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

- **isCancelRequested**
  - boolean, True if run cancel requested, else false.

- **startedAt**
  - string, The time the last run started at.

- **finishedAt**
  - string, The time the last run completed.

- **error**
  - string, The error, if any, returned by the run.
Get the logs for a run

**scripts_list_python3_runs_logs**

**Description**

Get the logs for a run

**Usage**

```python
scripts_list_python3_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

**Arguments**

- **id**: integer required. The ID of the python.
- **run_id**: integer required. The ID of the run.
- **last_id**: integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted. Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
- **limit**: integer optional. The maximum number of log messages to return. Default of 10000.

**Value**

An array containing the following fields:

- **id**: integer, The ID of the log.
- **createdAt**: string, The time the log was created.
- **message**: string, The log message.
- **level**: string, The level of the log. One of unknown, fatal, error, warn, info, debug.

---

List the outputs for a run

**scripts_list_python3_runs_outputs**

**Description**

List the outputs for a run
Usage

```python
scripts_list_python3_runs_outputs(
    id,
    run_id,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```

Arguments

- **id**
  - integer required. The ID of the python script.
- **run_id**
  - integer required. The ID of the run.
- **limit**
  - integer optional. Number of results to return. Defaults to its maximum of 50.
- **page_num**
  - integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order**
  - string optional. The field on which to order the result set. Defaults to `created_at`. Must be one of: `created_at`, `id`.
- **order_dir**
  - string optional. Direction in which to sort, either `asc` (ascending) or `desc` (descending) defaulting to `desc`.

Value

An array containing the following fields:

- **objectType**
  - string, The type of the output. Valid values are `File`, `Table`, `Report`, `Project`, `Credential`, or `JSONValue`
- **objectId**
  - integer, The ID of the output.
- **name**
  - string, The name of the output.
- **link**
  - string, The hypermedia link to the output.
- **value**
  - string,

---

**scripts_list_python3_shares**

*List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

```python
scripts_list_python3_shares(id)
```
The script `scripts_list_r_git` is used to get the git metadata attached to an item.

**Description**

Get the git metadata attached to an item

**Usage**

```bash
scripts_list_r_git(id)
```

**Arguments**

- **id**
  - integer required. The ID of the file.
Value

A list containing the following elements:

- **gitRef**: string, A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
- **gitBranch**: string, The git branch that the file is on.
- **gitPath**: string, The path of the file in the repository.
- **gitRepo**: list, A list containing the following elements:
  - **id**: integer, The ID for this git repository.
  - **repoUrl**: string, The URL for this git repository.
  - **createdAt**: string,
  - **updatedAt**: string,
- **pullFromGit**: boolean, Automatically pull latest commit from git. Only works for scripts.

---

**scripts_list_r_git_commits**

*Get the git commits for an item*

---

Description

Get the git commits for an item

Usage

```
scripts_list_r_git_commits(id)
```

Arguments

- **id**: integer required. The ID of the file.

Value

A list containing the following elements:

- **commitHash**: string, The SHA of the commit.
- **authorName**: string, The name of the commit’s author.
- **date**: string, The commit’s timestamp.
- **message**: string, The commit message.
scripts_list_r_projects

List the projects an R Script belongs to

Description

List the projects an R Script belongs to

Usage

scripts_list_r_projects(id, hidden = NULL)

Arguments

id integer required. The ID of the R Script.
hidden boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

id integer, The ID for this project.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
name string, The name of this project.
description string, A description of the project.
users array, An array containing the following fields:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
autoShare boolean,
createdAt string,
updatedAt string,
archived string, The archival status of the requested item(s).
scripts_list_r_runs

List runs for the given r

Description
List runs for the given r

Usage
scripts_list_r_runs(id, limit = NULL, page_num = NULL, order = NULL, order_dir = NULL)

Arguments
id integer required. The ID of the r.
limit integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num integer optional. Page number of the results to return. Defaults to the first page, 1.
order string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value
An array containing the following fields:
id integer, The ID of the run.
rid integer, The ID of the r.
state string, The state of the run, one of ‘queued’ ‘running’ ‘succeeded’ ‘failed’ or ‘cancelled’.
isCancelRequested boolean, True if run cancel requested, else false.
startedAt string, The time the last run started at.
finishedAt string, The time the last run completed.
error string, The error, if any, returned by the run.
scripts_list_r_runs_logs

*Get the logs for a run*

**Description**

Get the logs for a run

**Usage**

```r
scripts_list_r_runs_logs(id, run_id, last_id = NULL, limit = NULL)
```

**Arguments**

- **id**  
  integer required. The ID of the r.
- **run_id**  
  integer required. The ID of the run.
- **last_id**  
  integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted. Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
- **limit**  
  integer optional. The maximum number of log messages to return. Default of 10000.

**Value**

An array containing the following fields:

- **id** integer, The ID of the log.
- **createdAt** string, The time the log was created.
- **message** string, The log message.
- **level** string, The level of the log. One of unknown, fatal, error, warn, info, debug.

---

scripts_list_r_runs_outputs

*List the outputs for a run*

**Description**

List the outputs for a run
usage

scripts_list_r_runs_outputs(
    id,
    run_id,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)

arguments

id integer required. The ID of the r script.
run_id integer required. The ID of the run.
limit integer optional. Number of results to return. Defaults to its maximum of 50.
page_num integer optional. Page number of the results to return. Defaults to the first page, 1.
order string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
order_dir string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

value

An array containing the following fields:

objectType string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId integer, The ID of the output.
name string, The name of the output.
link string, The hypermedia link to the output.
value string,

scripts_list_r_shares List users and groups permissioned on this object

description

List users and groups permissioned on this object

usage

scripts_list_r_shares(id)
**Arguments**

id  
integer required. The ID of the resource that is shared.

**Value**

An array containing the following fields:

**readers** list, A list containing the following elements:

- users array,
- groups array,

**writers** list, A list containing the following elements:

- users array,
- groups array,

**owners** list, A list containing the following elements:

- users array,
- groups array,

**totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

**totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**scripts_list_sql_git**  
*Get the git metadata attached to an item*

---

**Description**

Get the git metadata attached to an item

**Usage**

`scripts_list_sql_git(id)`

**Arguments**

id  
integer required. The ID of the file.
Value

A list containing the following elements:

- `gitRef` string, A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
- `gitBranch` string, The git branch that the file is on.
- `gitPath` string, The path of the file in the repository.
- `gitRepo` list, A list containing the following elements:
  - `id` integer, The ID for this git repository.
  - `repoUrl` string, The URL for this git repository.
  - `createdAt` string,
  - `updatedAt` string,
- `pullFromGit` boolean, Automatically pull latest commit from git. Only works for scripts.

---

`scripts_list_sql_git_commits`

_Get the git commits for an item_

**Description**

Get the git commits for an item

**Usage**

`scripts_list_sql_git_commits(id)`

**Arguments**

- `id` integer required. The ID of the file.

**Value**

A list containing the following elements:

- `commitHash` string, The SHA of the commit.
- `authorName` string, The name of the commit’s author.
- `date` string, The commit’s timestamp.
- `message` string, The commit message.
scripts_list_sql_projects

List the projects a SQL script belongs to

Description
List the projects a SQL script belongs to

Usage
scripts_list_sql_projects(id, hidden = NULL)

Arguments
id integer required. The ID of the SQL script.
hidden boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value
An array containing the following fields:

id integer, The ID for this project.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
name string, The name of this project.
description string, A description of the project.
users array, An array containing the following fields:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
autoShare boolean,
createdAt string,
updatedAt string,
archived string, The archival status of the requested item(s).
scripts_list_sql_runs  List runs for the given sql

Description
List runs for the given sql

Usage
scripts_list_sql_runs(
  id,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)

Arguments
id  integer required. The ID of the sql.
limit  integer optional. Number of results to return. Defaults to 20. Maximum allowed is 100.
page_num  integer optional. Page number of the results to return. Defaults to the first page, 1.
order  string optional. The field on which to order the result set. Defaults to id. Must be one of: id.
order_dir  string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value
An array containing the following fields:
 id  integer, The ID of this run.
 sqlId  integer, The ID of this sql.
 state  string, The state of this run.
 isCancelRequested  boolean, True if run cancel requested, else false.
 startedAt  string, The time the last run started.
 finishedAt  string, The time that this run finished.
 error  string, The error message for this run, if present.
 output  array, An array containing the following fields:
  • outputName string, The name of the output file.
  • fileId integer, The unique ID of the output file.
  • path string, The temporary link to download this output file, valid for 36 hours.
scripts_list_sql_runs_logs

Get the logs for a run

Description

Get the logs for a run

Usage

scripts_list_sql_runs_logs(id, run_id, last_id = NULL, limit = NULL)

Arguments

- **id**: integer required. The ID of the SQL.
- **run_id**: integer required. The ID of the run.
- **last_id**: integer optional. The ID of the last log message received. Log entries with this ID value or lower will be omitted. Logs are sorted by ID if this value is provided, and are otherwise sorted by createdAt.
- **limit**: integer optional. The maximum number of log messages to return. Default of 10000.

Value

An array containing the following fields:

- **id**: integer, The ID of the log.
- **createdAt**: string, The time the log was created.
- **message**: string, The log message.
- **level**: string, The level of the log. One of unknown, fatal, error, warn, info, debug.

scripts_list_sql_runs_outputs

List the outputs for a run

Description

List the outputs for a run
Usage

scripts_list_sql_runs_outputs(
    id,
    run_id,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)

Arguments

  id          integer required. The ID of the sql script.
  run_id      integer required. The ID of the run.
  limit       integer optional. Number of results to return. Defaults to its maximum of 50.
  page_num    integer optional. Page number of the results to return. Defaults to the first page, 1.
  order       string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, id.
  order_dir   string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

  objectType  string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
  objectId   integer, The ID of the output.
  name       string, The name of the output.
  link       string, The hypermedia link to the output.
  value      string,

scripts_list_sql_shares

List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

scripts_list_sql_shares(id)
Arguments

id  integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers  list, A list containing the following elements:
  • users array,
  • groups array,

writers  list, A list containing the following elements:
  • users array,
  • groups array,

owners  list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares  integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares  integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

scripts_list_types  List available script types

Description

List available script types

Usage

scripts_list_types()

Value

An array containing the following fields:

name  string, The name of the type.
scripts_patch

Update a script

Description

Update a script

Usage

scripts_patch(
  id,
  name = NULL,
  sql = NULL,
  params = NULL,
  arguments = NULL,
  template_script_id = NULL,
  schedule = NULL,
  notifications = NULL,
  parent_id = NULL
)

Arguments

id        integer required. The ID for the script.
name      string optional. The name of the script.
sql       string optional. The raw SQL query for the script.
params    array optional. An array containing the following fields:
  • name string, The variable’s name as used within your code.
  • label string, The label to present to users when asking them for the value.
  • description string, A short sentence or fragment describing this parameter to the end user.
  • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  • required boolean, Whether this param is required.
  • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’
arguments

list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

template_script_id

integer optional. The ID of the template script, if any. A script cannot both have a template script and be a template for other scripts.

schedule

list optional. A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications

list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

parent_id

integer optional. The ID of the parent job that will trigger this script

Value

A list containing the following elements:

id

integer, The ID for the script.

name

string, The name of the script.

type

string, The type of script.

createdAt

string, The time this script was created.

updatedAt

string, The time this script was last updated.

author

list, A list containing the following elements:

- id integer, The ID of this user.
- name string, This user’s name.
- username string, This user’s username.
- initials string, This user’s initials.
- online boolean, Whether this user is online.

**state** string, The status of the script’s last run.

**finishedAt** string, The time that the script’s last run finished.

**category** string, The category of the script.

**projects** array, An array containing the following fields:
- id integer, The ID for the project.
- name string, The name of the project.

**parentId** integer, The ID of the parent job that will trigger this script.

**userContext** string, "runner" or "author", who to execute the script as when run as a template.

**params** array, An array containing the following fields:
- name string, The variable’s name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’

**arguments** list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

**isTemplate** boolean, Whether others scripts use this one as a template.

**publishedAsTemplateId** integer, The ID of the template that this script is backing.

**fromTemplateId** integer, The ID of the template this script uses, if any.

**templateDependentsCount** integer, How many other scripts use this one as a template.

**templateScriptName** string, The name of the template script.

**links** list, A list containing the following elements:
- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.
scripts_patch

schedule list, A list containing the following elements:
- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
- id integer, The ID of this user.
- name string, This user’s name.
- username string, This user’s username.
- initials string, This user’s initials.
- online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.
timeZone string, The time zone of this script.
lastRun list, A list containing the following elements:
- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.
targetProjectId integer, Target project to which script outputs will be added.
scripts_patch_containers

Update a container

Description
Update a container

Usage

scripts_patch_containers(
    id,
    name = NULL,
    parent_id = NULL,
    user_context = NULL,
    params = NULL,
    arguments = NULL,
    schedule = NULL,
    notifications = NULL,
    required_resources = NULL,
    repo_http_uri = NULL,
    repo_ref = NULL,
    remote_host_credential_id = NULL,
    git_credential_id = NULL,
    docker_command = NULL,
    docker_image_name = NULL,
    docker_image_tag = NULL,
    instance_type = NULL,
    cancel_timeout = NULL,
    time_zone = NULL,
    target_project_id = NULL
)

Arguments

id integer required. The ID for the script.
name string optional. The name of the container.
parent_id integer optional. The ID of the parent job that will trigger this script.
user_context  string optional. "runner" or "author", who to execute the script as when run as a template.

params  array optional. An array containing the following fields:
  • name string. The variable’s name as used within your code.
  • label string. The label to present to users when asking them for the value.
  • description string. A short sentence or fragment describing this parameter to the end user.
  • type string. The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  • required boolean. Whether this param is required.
  • value string. The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  • default string. If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  • allowedValues array. The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: 'label: 'Import', 'value': 'import''

arguments  list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

schedule  list optional. A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications  list optional. A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string. Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • successEmailFromName string, Name from which success emails are sent; defaults to “Civis.”
  • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
scripts_patch_containers

- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

required_resources

list optional. A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares.
- memory integer, The amount of RAM to allocate for the container (in MB).
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.
- wholeInstance boolean, Whether or not to use the entire instance. If true, cpu, memory, and disk space are not required and will be set to an instance’s max.

repo_http_uri string optional. The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

repo_ref string optional. The tag or branch of the github repo to clone into the container.

remote_host_credential_id integer optional. The id of the database credentials to pass into the environment of the container.

git_credential_id integer optional. The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you’ve submitted will be used. Unnecessary if no git repo is specified or the git repo is public.

docker_command string optional. The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]. Defaults to the Docker image’s ENTRY-POINT/CMD.

docker_image_name string optional. The name of the docker image to pull from DockerHub.

docker_image_tag string optional. The tag of the docker image to pull from DockerHub.

instance_type string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

cancel_timeout integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

time_zone string optional. The time zone of this script.

target_project_id integer optional. Target project to which script outputs will be added.

Value

A list containing the following elements:

id integer, The ID for the script.
### scripts_patch_containers

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>string, The name of the container.</td>
</tr>
<tr>
<td>type</td>
<td>string, The type of the script (e.g Container)</td>
</tr>
<tr>
<td>createdAt</td>
<td>string, The time this script was created.</td>
</tr>
<tr>
<td>updatedAt</td>
<td>string, The time the script was last updated.</td>
</tr>
<tr>
<td>author</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• id integer, The ID of this user.</td>
</tr>
<tr>
<td></td>
<td>• name string, This user’s name.</td>
</tr>
<tr>
<td></td>
<td>• username string, This user’s username.</td>
</tr>
<tr>
<td></td>
<td>• initials string, This user’s initials.</td>
</tr>
<tr>
<td></td>
<td>• online boolean, Whether this user is online.</td>
</tr>
<tr>
<td>state</td>
<td>string, The status of the script’s last run.</td>
</tr>
<tr>
<td>finishedAt</td>
<td>string, The time that the script’s last run finished.</td>
</tr>
<tr>
<td>category</td>
<td>string, The category of the script.</td>
</tr>
<tr>
<td>projects</td>
<td>array, An array containing the following fields:</td>
</tr>
<tr>
<td></td>
<td>• id integer, The ID for the project.</td>
</tr>
<tr>
<td></td>
<td>• name string, The name of the project.</td>
</tr>
<tr>
<td>parentId</td>
<td>integer, The ID of the parent job that will trigger this script</td>
</tr>
<tr>
<td>userContext</td>
<td>string, &quot;runner&quot; or &quot;author&quot;, who to execute the script as when run as a template.</td>
</tr>
<tr>
<td>params</td>
<td>array, An array containing the following fields:</td>
</tr>
<tr>
<td></td>
<td>• name string, The variable’s name as used within your code.</td>
</tr>
<tr>
<td></td>
<td>• label string, The label to present to users when asking them for the value.</td>
</tr>
<tr>
<td></td>
<td>• description string, A short sentence or fragment describing this parameter to the end user.</td>
</tr>
<tr>
<td></td>
<td>• type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom</td>
</tr>
<tr>
<td></td>
<td>• required boolean, Whether this param is required.</td>
</tr>
<tr>
<td></td>
<td>• value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.</td>
</tr>
<tr>
<td></td>
<td>• default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.</td>
</tr>
<tr>
<td></td>
<td>• allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’ ‘</td>
</tr>
<tr>
<td>arguments</td>
<td>list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.</td>
</tr>
<tr>
<td>isTemplate</td>
<td>boolean, Whether others scripts use this one as a template.</td>
</tr>
<tr>
<td>templateDependentsCount</td>
<td>integer, How many other scripts use this one as a template.</td>
</tr>
</tbody>
</table>
scripts_patch_containers

**publishedAsTemplateId**
integer, The ID of the template that this script is backing.

**fromTemplateId**
integer, The ID of the template script.

**templateScriptName**
string, The name of the template script.

**links**
list, A list containing the following elements:
- **details** string, The details link to get more information about the script.
- **runs** string, The runs link to get the run information list for this script.

**schedule**
list, A list containing the following elements:
- **scheduled** boolean, If the item is scheduled.
- **scheduledDays** array, Day based on numeric value starting at 0 for Sunday.
- **scheduledHours** array, Hours of the day it is scheduled on.
- **scheduledMinutes** array, Minutes of the day it is scheduled on.
- **scheduledRunsPerHour** integer, Alternative to scheduled minutes, number of times to run per hour.

**notifications**
list, A list containing the following elements:
- **urls** array, URLs to receive a POST request at job completion
- **successEmailSubject** string, Custom subject line for success e-mail.
- **successEmailBody** string, Custom body text for success e-mail, written in Markdown.
- **successEmailAddresses** array, Addresses to notify by e-mail when the job completes successfully.
- **successEmailFromName** string, Name from which success emails are sent; defaults to "Civis."
- **successEmailReplyTo** string, Address for replies to success emails; defaults to the author of the job.
- **failureEmailAddresses** array, Addresses to notify by e-mail when the job fails.
- **stallWarningMinutes** integer, Stall warning emails will be sent after this amount of minutes.
- **successOn** boolean, If success email notifications are on.
- **failureOn** boolean, If failure email notifications are on.

**runningAs**
list, A list containing the following elements:
- **id** integer, The ID of this user.
- **name** string, This user’s name.
- **username** string, This user’s username.
- **initials** string, This user’s initials.
- **online** boolean, Whether this user is online.

**requiredResources**
list, A list containing the following elements:
- **cpu** integer, The number of CPU shares to allocate for the container. Each core has 1000 shares.
* memory integer, The amount of RAM to allocate for the container (in MB).
* diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

** repoHttpUri string, The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

** repoRef string, The tag or branch of the github repo to clone into the container.

** remoteHostCredentialId integer, The id of the database credentials to pass into the environment of the container.

** gitCredentialId integer, The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you’ve submitted will be used. Unnecessary if no git repo is specified or the git repo is public.

** dockerCommand string, The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]. Defaults to the Docker image’s ENTRYPOINT/CMD.

** dockerImageName string, The name of the docker image to pull from DockerHub.

** dockerImageTag string, The tag of the docker image to pull from DockerHub.

** instanceType string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

** cancelTimeout integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

** lastRun list, A list containing the following elements:
  * id integer,
  * state string,
  * createdAt string, The time that the run was queued.
  * startedAt string, The time that the run started.
  * finishedAt string, The time that the run completed.
  * error string, The error message for this run, if present.

** timeZone string, The time zone of this script.

** hidden boolean, The hidden status of the item.

** archived string, The archival status of the requested item(s).

** targetProjectId integer, Target project to which script outputs will be added.
scripts\_patch\_custom  
*Update some attributes of this Custom Script*

**Description**

Update some attributes of this Custom Script

**Usage**

```r
scripts\_patch\_custom(
    id,
    name = NULL,
    parent_id = NULL,
    arguments = NULL,
    remote_host_id = NULL,
    credential_id = NULL,
    schedule = NULL,
    notifications = NULL,
    time_zone = NULL,
    target_project_id = NULL
)
```

**Arguments**

- **id**  
  integer required. The ID for the script.

- **name**  
  string optional. The name of the script.

- **parent_id**  
  integer optional. The ID of the parent job that will trigger this script

- **arguments**  
  list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

- **remote_host_id**  
  integer optional. The remote host ID that this script will connect to.

- **credential_id**  
  integer optional. The credential that this script will use.

- **schedule**  
  list optional. A list containing the following elements:
  - **scheduled**  
    boolean, If the item is scheduled.
  - **scheduledDays**  
    array, Day based on numeric value starting at 0 for Sunday.
  - **scheduledHours**  
    array, Hours of the day it is scheduled on.
  - **scheduledMinutes**  
    array, Minutes of the day it is scheduled on.
  - **scheduledRunsPerHour**  
    integer, Alternative to scheduled minutes, number of times to run per hour.

- **notifications**  
  list optional. A list containing the following elements:
  - **urls**  
    array, URLs to receive a POST request at job completion
  - **successEmailSubject**  
    string, Custom subject line for success e-mail.
  - **successEmailBody**  
    string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

time_zone
string optional. The time zone of this script.
target_project_id
integer optional. Target project to which script outputs will be added.

Value

A list containing the following elements:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer, The ID for the script.</td>
</tr>
<tr>
<td>name</td>
<td>string, The name of the script.</td>
</tr>
<tr>
<td>type</td>
<td>string, The type of the script (e.g Custom)</td>
</tr>
<tr>
<td>createdAt</td>
<td>string, The time this script was created.</td>
</tr>
<tr>
<td>updatedAt</td>
<td>string, The time the script was last updated.</td>
</tr>
<tr>
<td>author</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• id integer, The ID of this user.</td>
</tr>
<tr>
<td></td>
<td>• name string, This user’s name.</td>
</tr>
<tr>
<td></td>
<td>• username string, This user’s username.</td>
</tr>
<tr>
<td></td>
<td>• initials string, This user’s initials.</td>
</tr>
<tr>
<td></td>
<td>• online boolean, Whether this user is online.</td>
</tr>
<tr>
<td>state</td>
<td>string, The status of the script’s last run.</td>
</tr>
<tr>
<td>finishedAt</td>
<td>string, The time that the script’s last run finished.</td>
</tr>
<tr>
<td>category</td>
<td>string,</td>
</tr>
<tr>
<td>projects</td>
<td>array, An array containing the following fields:</td>
</tr>
<tr>
<td></td>
<td>• id integer, The ID for the project.</td>
</tr>
<tr>
<td></td>
<td>• name string, The name of the project.</td>
</tr>
<tr>
<td>parentId</td>
<td>integer, The ID of the parent job that will trigger this script</td>
</tr>
<tr>
<td>params</td>
<td>array, An array containing the following fields:</td>
</tr>
<tr>
<td></td>
<td>• name string, The variable’s name as used within your code.</td>
</tr>
<tr>
<td></td>
<td>• label string, The label to present to users when asking them for the value.</td>
</tr>
<tr>
<td></td>
<td>• description string, A short sentence or fragment describing this parameter to the end user.</td>
</tr>
</tbody>
</table>
• **type** string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom

• **required** boolean, Whether this param is required.

• **value** string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.

• **default** string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.

• **allowedValues** array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’

**arguments**

list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

**isTemplate**

boolean, Whether others scripts use this one as a template.

**publishedAsTemplateId**

integer, The ID of the template that this script is backing.

**fromTemplateId**

integer, The ID of the template script.

**uiReportUrl**

integer, The url of the custom HTML.

**uiReportId**

integer, The id of the report with the custom HTML.

**uiReportProvideAPIKey**

boolean, Whether the ui report requests an API Key from the report viewer.

**templateScriptName**

string, The name of the template script.

**templateNote**

string, The template’s note.

**remoteHostId**

integer, The remote host ID that this script will connect to.

**credentialId**

integer, The credential that this script will use.

**codePreview**

string, The code that this script will run with arguments inserted.

**schedule**

list, A list containing the following elements:

• **scheduled** boolean, If the item is scheduled.

• **scheduledDays** array, Day based on numeric value starting at 0 for Sunday.

• **scheduledHours** array, Hours of the day it is scheduled on.

• **scheduledMinutes** array, Minutes of the day it is scheduled on.

• **scheduledRunsPerHour** integer, Alternative to scheduled minutes, number of times to run per hour.

**notifications**

list, A list containing the following elements:

• **urls** array, URLs to receive a POST request at job completion

• **successEmailSubject** string, Custom subject line for success e-mail.

• **successEmailBody** string, Custom body text for success e-mail, written in Markdown.

• **successEmailAddresses** array, Addresses to notify by e-mail when the job completes successfully.
- **successEmailFromName** string, Name from which success emails are sent; defaults to "Civis."
- **successEmailReplyTo** string, Address for replies to success emails; defaults to the author of the job.
- **failureEmailAddresses** array, Addresses to notify by e-mail when the job fails.
- **stallWarningMinutes** integer, Stall warning emails will be sent after this amount of minutes.
- **successOn** boolean, If success email notifications are on.
- **failureOn** boolean, If failure email notifications are on.

**runningAs** list, A list containing the following elements:

- **id** integer, The ID of this user.
- **name** string, This user’s name.
- **username** string, This user’s username.
- **initials** string, This user’s initials.
- **online** boolean, Whether this user is online.

**timeZone** string, The time zone of this script.

**lastRun** list, A list containing the following elements:

- **id** integer,
- **state** string,
- **createdAt** string, The time that the run was queued.
- **startedAt** string, The time that the run started.
- **finishedAt** string, The time that the run completed.
- **error** string, The error message for this run, if present.

**hidden** boolean, The hidden status of the item.

**archived** string, The archival status of the requested item(s).

**targetProjectId** integer, Target project to which script outputs will be added.

**lastSuccessfulRun** list, A list containing the following elements:

- **id** integer,
- **state** string,
- **createdAt** string, The time that the run was queued.
- **startedAt** string, The time that the run started.
- **finishedAt** string, The time that the run completed.
- **error** string, The error message for this run, if present.
scripts_patch_javascript

*Update some attributes of this JavaScript Script*

**Description**

Update some attributes of this JavaScript Script

**Usage**

```python
scripts_patch_javascript(
    id,
    name = NULL,
    parent_id = NULL,
    user_context = NULL,
    params = NULL,
    arguments = NULL,
    schedule = NULL,
    notifications = NULL,
    next_run_at = NULL,
    time_zone = NULL,
    target_project_id = NULL,
    source = NULL,
    remote_host_id = NULL,
    credential_id = NULL
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID for the script.</td>
</tr>
<tr>
<td>name</td>
<td>string optional. The name of the script.</td>
</tr>
<tr>
<td>parent_id</td>
<td>integer optional. The ID of the parent job that will trigger this script</td>
</tr>
<tr>
<td>user_context</td>
<td>string optional. &quot;runner&quot; or &quot;author&quot;, who to execute the script as when run as a template.</td>
</tr>
<tr>
<td>params</td>
<td>array optional. An array containing the following fields:</td>
</tr>
<tr>
<td></td>
<td>• name string. The variable’s name as used within your code.</td>
</tr>
<tr>
<td></td>
<td>• label string. The label to present to users when asking them for the value.</td>
</tr>
<tr>
<td></td>
<td>• description string. A short sentence or fragment describing this parameter to the end user.</td>
</tr>
<tr>
<td></td>
<td>• type string. The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom</td>
</tr>
<tr>
<td></td>
<td>• required boolean, Whether this param is required.</td>
</tr>
<tr>
<td></td>
<td>• value string. The value you would like to set this param to. Setting this value makes this parameter a fixed param.</td>
</tr>
</tbody>
</table>
• default string. If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.

• allowedValues array. The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: 'Import', 'value': 'import’

arguments: list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

schedule: list optional. A list containing the following elements:
  • scheduled boolean. If the item is scheduled.
  • scheduledDays array. Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array. Hours of the day it is scheduled on.
  • scheduledMinutes array. Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer. Alternative to scheduled minutes, number of times to run per hour.

notifications: list optional. A list containing the following elements:
  • urls array. URLs to receive a POST request at job completion
  • successEmailSubject string. Custom subject line for success e-mail.
  • successEmailBody string. Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array. Addresses to notify by e-mail when the job completes successfully.
  • successEmailFromName string. Name from which success emails are sent; defaults to "Civis."
  • successEmailReplyTo string. Address for replies to success emails; defaults to the author of the job.
  • failureEmailAddresses array. Addresses to notify by e-mail when the job fails.
  • stallWarningMinutes integer. Stall warning emails will be sent after this amount of minutes.
  • successOn boolean. If success email notifications are on.
  • failureOn boolean. If failure email notifications are on.

next_run_at: string optional. The time of the next scheduled run.
time_zone: string optional. The time zone of this script.
target_project_id: integer optional. Target project to which script outputs will be added.
source: string optional. The body/text of the script.
remote_host_id: integer optional. The remote host ID that this script will connect to.
credential_id: integer optional. The credential that this script will use.
Value

A list containing the following elements:

- **id** integer, The ID for the script.
- **name** string, The name of the script.
- **type** string, The type of the script (e.g SQL, Container, Python, R, JavaScript)
- **createdAt** string, The time this script was created.
- **updatedAt** string, The time the script was last updated.
- **author** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **state** string, The status of the script’s last run.
- **finishedAt** string, The time that the script’s last run finished.
- **category** string, The category of the script.
- **projects** array, An array containing the following fields:
  - **id** integer, The ID for the project.
  - **name** string, The name of the project.
- **parentId** integer, The ID of the parent job that will trigger this script
- **userContext** string, "runner" or "author", who to execute the script as when run as a template.
- **params** array, An array containing the following fields:
  - **name** string, The variable’s name as used within your code.
  - **label** string, The label to present to users when asking them for the value.
  - **description** string, A short sentence or fragment describing this parameter to the end user.
  - **type** string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  - **required** boolean, Whether this param is required.
  - **value** string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  - **default** string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  - **allowedValues** array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’
- **arguments** list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
isTemplate boolean, Whether others scripts use this one as a template.
publishedAsTemplateId integer, The ID of the template that this script is backing.
fromTemplateId integer, The ID of the template this script uses, if any.
templateDependentsCount integer, How many other scripts use this one as a template.
templateScriptName string, The name of the template script.

links list, A list containing the following elements:
  • details string, The details link to get more information about the script.
  • runs string, The runs link to get the run information list for this script.

schedule list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  • successOn boolean, If success email notifications are on.
  • failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.
timeZone

string, The time zone of this script.

lastRun

list, A list containing the following elements:

- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

hidden

boolean, The hidden status of the item.

targetProjectId

integer, Target project to which script outputs will be added.

archived

string, The archival status of the requested item(s).

source

string, The body/text of the script.

remoteHostId

integer, The remote host ID that this script will connect to.

credentialId

integer, The credential that this script will use.

---

scripts_patch_python3  
*Update some attributes of this Python Script*

### Description

Update some attributes of this Python Script

### Usage

```python
def scripts_patch_python3(
    id,
    name = NULL,
    parent_id = NULL,
    user_context = NULL,
    params = NULL,
    arguments = NULL,
    schedule = NULL,
    notifications = NULL,
    next_run_at = NULL,
    time_zone = NULL,
    target_project_id = NULL,
    required_resources = NULL,
    instance_type = NULL,
    source = NULL,
    cancel_timeout = NULL,
    docker_image_tag = NULL
)
```
Arguments

id integer required. The ID for the script.
name string optional. The name of the script.
parent_id integer optional. The ID of the parent job that will trigger this script
user_context string optional. "runner" or "author", who to execute the script as when run as a template.
params array optional. An array containing the following fields:
  • name string. The variable’s name as used within your code.
  • label string. The label to present to users when asking them for the value.
  • description string. A short sentence or fragment describing this parameter to the end user.
  • type string. The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  • required boolean. Whether this param is required.
  • value string. The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  • default string. If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  • allowedValues array. The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ’Import’, ’value’: ’import’
arguments list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
schedule list optional. A list containing the following elements:
  • scheduled boolean. If the item is scheduled.
  • scheduledDays array. Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array. Hours of the day it is scheduled on.
  • scheduledMinutes array. Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer. Alternative to scheduled minutes, number of times to run per hour.
notifications list optional. A list containing the following elements:
  • urls array. URLs to receive a POST request at job completion
  • successEmailSubject string. Custom subject line for success e-mail.
  • successEmailBody string. Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array. Addresses to notify by e-mail when the job completes successfully.
  • successEmailFromName string. Name from which success emails are sent; defaults to "Civis."
scripts_patch_python3

- successEmailReplyTo string. Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array. Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer. Stall warning emails will be sent after this amount of minutes.
- successOn boolean. If success email notifications are on.
- failureOn boolean. If failure email notifications are on.

next_run_at string optional. The time of the next scheduled run.
time_zone string optional. The time zone of this script.
target_project_id integer optional. Target project to which script outputs will be added.

required_resources
list optional. A list containing the following elements:
- cpu integer. The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer. The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number. The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

instance_type string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.
source string optional. The body/text of the script.
cancel_timeout integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.
docker_image_tag
string optional. The tag of the docker image to pull from DockerHub.

Value

A list containing the following elements:

id integer, The ID for the script.
name string, The name of the script.
type string, The type of the script (e.g SQL, Container, Python, R, JavaScript)
createdAt string, The time this script was created.
updatedAt string, The time the script was last updated.
author list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

state string, The status of the script’s last run.
finishedAt string, The time that the script’s last run finished.
category string, The category of the script.
projects array, An array containing the following fields:
  • id integer, The ID for the project.
  • name string, The name of the project.

parentId integer, The ID of the parent job that will trigger this script
userContext string, "runner" or "author", who to execute the script as when run as a template.

params array, An array containing the following fields:
  • name string, The variable’s name as used within your code.
  • label string, The label to present to users when asking them for the value.
  • description string, A short sentence or fragment describing this parameter to the end user.
  • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  • required boolean, Whether this param is required.
  • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
isTemplate boolean, Whether others scripts use this one as a template.
publishedAsTemplateId integer, The ID of the template that this script is backing.
fromTemplateId integer, The ID of the template this script uses, if any.
templateDependentsCount integer, How many other scripts use this one as a template.
templateScriptName string, The name of the template script.

links list, A list containing the following elements:
  • details string, The details link to get more information about the script.
  • runs string, The runs link to get the run information list for this script.
schedule  list, A list containing the following elements:
   • scheduled boolean, If the item is scheduled.
   • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
   • scheduledHours array, Hours of the day it is scheduled on.
   • scheduledMinutes array, Minutes of the day it is scheduled on.
   • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
   • urls array, URLs to receive a POST request at job completion
   • successEmailSubject string, Custom subject line for success e-mail.
   • successEmailBody string, Custom body text for success e-mail, written in Markdown.
   • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
   • successEmailFromName string, Name from which success emails are sent; defaults to “Civis.”
   • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
   • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
   • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
   • successOn boolean, If success email notifications are on.
   • failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
   • id integer, The ID of this user.
   • name string, This user’s name.
   • username string, This user’s username.
   • initials string, This user’s initials.
   • online boolean, Whether this user is online.

timeZone string, The time zone of this script.

nextRunAt string, The time of the next scheduled run.
lastRun list, A list containing the following elements:
   • id integer,
   • state string,
   • createdAt string, The time that the run was queued.
   • startedAt string, The time that the run started.
   • finishedAt string, The time that the run completed.
   • error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.
targetProjectId integer, Target project to which script outputs will be added.
archives: string, The archival status of the requested item(s).

requiredResources: list, A list containing the following elements:

- cpu: integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory: integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace: number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

instanceType: string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

source: string, The body/text of the script.

cancelTimeout: integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

dockerImageTag: string, The tag of the docker image to pull from DockerHub.

---

**scripts_patch_r**

Update some attributes of this R Script

**Description**

Update some attributes of this R Script

**Usage**

```r
scripts_patch_r(
  id,
  name = NULL,
  parent_id = NULL,
  user_context = NULL,
  params = NULL,
  arguments = NULL,
  schedule = NULL,
  notifications = NULL,
  next_run_at = NULL,
  time_zone = NULL,
  target_project_id = NULL,
  required_resources = NULL,
  instance_type = NULL,
  source = NULL,
  cancel_timeout = NULL,
  docker_image_tag = NULL
)
```
Arguments

- **id**
  - integer required. The ID for the script.

- **name**
  - string optional. The name of the script.

- **parent_id**
  - integer optional. The ID of the parent job that will trigger this script.

- **user_context**
  - string optional. "runner" or "author", who to execute the script as when run as a template.

- **params**
  - array optional. An array containing the following fields:
    - name string. The variable’s name as used within your code.
    - label string. The label to present to users when asking them for the value.
    - description string. A short sentence or fragment describing this parameter to the end user.
    - type string. The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
    - required boolean. Whether this param is required.
    - value string. The value you would like to set this param to. Setting this value makes this parameter a fixed param.
    - default string. If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
    - allowedValues array. The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: 'Import', ‘value': 'import'”

- **arguments**
  - list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

- **schedule**
  - list optional. A list containing the following elements:
    - scheduled boolean. If the item is scheduled.
    - scheduledDays array, Day based on numeric value starting at 0 for Sunday.
    - scheduledHours array, Hours of the day it is scheduled on.
    - scheduledMinutes array, Minutes of the day it is scheduled on.
    - scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

- **notifications**
  - list optional. A list containing the following elements:
    - urls array, URLs to receive a POST request at job completion
    - successEmailSubject string, Custom subject line for success e-mail.
    - successEmailBody string, Custom body text for success e-mail, written in Markdown.
    - successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
    - successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by email when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

next_run_at string optional. The time of the next scheduled run.
time_zone string optional. The time zone of this script.
target_project_id integer optional. Target project to which script outputs will be added.
required_resources list optional. A list containing the following elements:
  • cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
  • memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
  • diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.
instance_type string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.
source string optional. The body/text of the script.
cancel_timeout integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.
docker_image_tag string optional. The tag of the docker image to pull from DockerHub.

Value
A list containing the following elements:

id integer, The ID for the script.
name string, The name of the script.
type string, The type of the script (e.g SQL, Container, Python, R, JavaScript)
createdAt string, The time this script was created.
updatedAt string, The time the script was last updated.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
• username string, This user's username.
• initials string, This user's initials.
• online boolean, Whether this user is online.

state string, The status of the script's last run.
finishedAt string, The time that the script's last run finished.
category string, The category of the script.
projects array, An array containing the following fields:
• id integer, The ID for the project.
• name string, The name of the project.

parentId integer, The ID of the parent job that will trigger this script
userContext string, "runner" or "author", who to execute the script as when run as a template.
params array, An array containing the following fields:
• name string, The variable's name as used within your code.
• label string, The label to present to users when asking them for the value.
• description string, A short sentence or fragment describing this parameter to the end user.
• type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
• required boolean, Whether this param is required.
• value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
• default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
• allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: 'label': 'Import', 'value': 'import'

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
isTemplate boolean, Whether others scripts use this one as a template.
publishedAsTemplateId integer, The ID of the template that this script is backing.
fromTemplateId integer, The ID of the template this script uses, if any.
templateDependentsCount integer, How many other scripts use this one as a template.
templateScriptName string, The name of the template script.

links list, A list containing the following elements:
• details string, The details link to get more information about the script.
• runs string, The runs link to get the run information list for this script.
scripts_patch_r

schedule list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number
    of times to run per hour.

notifications list, A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in
    Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job
    completes successfully.
  • successEmailFromName string, Name from which success emails are sent;
    defaults to "Civis."
  • successEmailReplyTo string, Address for replies to success emails; defaults
    to the author of the job.
  • failureEmailAddresses array, Addresses to notify by e-mail when the job
    fails.
  • stallWarningMinutes integer, Stall warning emails will be sent after this
    amount of minutes.
  • successOn boolean, If success email notifications are on.
  • failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.

timeZone string, The time zone of this script.

lastRun list, A list containing the following elements:
  • id integer,
  • state string,
  • createdAt string, The time that the run was queued.
  • startedAt string, The time that the run started.
  • finishedAt string, The time that the run completed.
  • error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.

targetProjectId integer, Target project to which script outputs will be added.
scripts_patch_sql

Update some attributes of this SQL script

Description

Update some attributes of this SQL script

Usage

scripts_patch_sql(
    id,
    name = NULL,
    parent_id = NULL,
    user_context = NULL,
    params = NULL,
    arguments = NULL,
    schedule = NULL,
    notifications = NULL,
    next_run_at = NULL,
    time_zone = NULL,
    target_project_id = NULL,
    sql = NULL,
    remote_host_id = NULL,
    credential_id = NULL,
    csv_settings = NULL
)
Arguments

- **id** integer required. The ID for the script.
- **name** string optional. The name of the script.
- **parent_id** integer optional. The ID of the parent job that will trigger this script
- **user_context** string optional. "runner" or "author", who to execute the script as when run as a template.
- **params** array optional. An array containing the following fields:
  - **name** string. The variable’s name as used within your code.
  - **label** string. The label to present to users when asking them for the value.
  - **description** string. A short sentence or fragment describing this parameter to the end user.
  - **type** string. The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  - **required** boolean. Whether this param is required.
  - **value** string. The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  - **default** string. If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  - **allowedValues** array. The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’
- **arguments** list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
- **schedule** list optional. A list containing the following elements:
  - **scheduled** boolean. If the item is scheduled.
  - **scheduledDays** array. Day based on numeric value starting at 0 for Sunday.
  - **scheduledHours** array. Hours of the day it is scheduled on.
  - **scheduledMinutes** array. Minutes of the day it is scheduled on.
  - **scheduledRunsPerHour** integer. Alternative to scheduled minutes, number of times to run per hour.
- **notifications** list optional. A list containing the following elements:
  - **urls** array. URLs to receive a POST request at job completion
  - **successEmailSubject** string. Custom subject line for success e-mail.
  - **successEmailBody** string. Custom body text for success e-mail, written in Markdown.
  - **successEmailAddresses** array. Addresses to notify by e-mail when the job completes successfully.
  - **successEmailFromName** string. Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

next_run_at string optional. The time of the next scheduled run.
time_zone string optional. The time zone of this script.
target_project_id integer optional. Target project to which script outputs will be added.
sql string optional. The raw SQL query for the script.
remote_host_id integer optional. The remote host ID that this script will connect to.
credential_id integer optional. The credential that this script will use.
csv_settings list optional. A list containing the following elements:
  • includeHeader boolean, Whether or not to include headers in the output data. Default: true
  • compression string, The type of compression to use, if any, one of "none", "zip", or "gzip". Default: gzip
  • columnDelimiter string, Which delimiter to use, one of "comma", "tab", or "pipe". Default: comma
  • unquoted boolean, Whether or not to quote fields. Default: false
  • forceMultifile boolean, Whether or not the csv should be split into multiple files. Default: false
  • filenamePrefix string, A user specified filename prefix for the output file to have. Default: null
  • maxFileSize integer, The max file size, in MB, created files will be. Only available when force_multifile is true.

Value

A list containing the following elements:

id integer, The ID for the script.
name string, The name of the script.
type string, The type of the script (e.g SQL, Container, Python, R, JavaScript)
createdAt string, The time this script was created.
updatedAt string, The time the script was last updated.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

state
string, The status of the script’s last run.

finishedAt
string, The time that the script’s last run finished.

category
string, The category of the script.

projects
array, An array containing the following fields:
• id integer, The ID for the project.
• name string, The name of the project.

parentId
integer, The ID of the parent job that will trigger this script

userContext
string, "runner" or "author", who to execute the script as when run as a template.

params
array, An array containing the following fields:
• name string, The variable’s name as used within your code.
• label string, The label to present to users when asking them for the value.
• description string, A short sentence or fragment describing this parameter to the end user.
• type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
• required boolean, Whether this param is required.
• value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
• default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
• allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: 'Import', 'value': 'import''

arguments
list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate
boolean, Whether others scripts use this one as a template.

publishedAsTemplateId
integer, The ID of the template that this script is backing.

fromTemplateId
integer, The ID of the template this script uses, if any.

templateDependentsCount
integer, How many other scripts use this one as a template.

templateScriptName
string, The name of the template script.

links
list, A list containing the following elements:
• details string, The details link to get more information about the script.
• runs string, The runs link to get the run information list for this script.

schedule
list, A list containing the following elements:
- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

**notifications** list, A list containing the following elements:
- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

**runningAs** list, A list containing the following elements:
- id integer, The ID of this user.
- name string, This user’s name.
- username string, This user’s username.
- initials string, This user’s initials.
- online boolean, Whether this user is online.

**nextRunAt** string, The time of the next scheduled run.

**timeZone** string, The time zone of this script.

**lastRun** list, A list containing the following elements:
- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.

targetProjectId integer, Target project to which script outputs will be added.

archived string, The archival status of the requested item(s).
scripts_post

Create a script

Description

Create a script

Usage

scripts_post(
    name,
    remote_host_id,
    credential_id,
    sql,
    params = NULL,
    arguments = NULL,
    template_script_id = NULL,
    notifications = NULL,
    hidden = NULL
)

sql string, The raw SQL query for the script.
expandedArguments list, Expanded arguments for use in injecting into different environments.
remoteHostId integer, The remote host ID that this script will connect to.
credentialId integer, The credential that this script will use.
codePreview string, The code that this script will run with arguments inserted.
csvSettings list, A list containing the following elements:
  • includeHeader boolean, Whether or not to include headers in the output data. Default: true
  • compression string, The type of compression to use, if any, one of "none", "zip", or "gzip". Default: gzip
  • columnDelimiter string, Which delimiter to use, one of "comma", "tab", or "pipe". Default: comma
  • unquoted boolean, Whether or not to quote fields. Default: false
  • forceMultifile boolean, Whether or not the csv should be split into multiple files. Default: false
  • filenamePrefix string, A user specified filename prefix for the output file to have. Default: null
  • maxFileSize integer, The max file size, in MB, created files will be. Only available when force_multifile is true.
Arguments

**name**  
string required. The name of the script.

**remote_host_id**  
integer required. The database ID.

**credential_id**  
integer required. The credential ID.

**sql**  
string required. The raw SQL query for the script.

**params**  
array optional. An array containing the following fields:
- **name** string. The variable’s name as used within your code.
- **label** string. The label to present to users when asking them for the value.
- **description** string. A short sentence or fragment describing this parameter to the end user.
- **type** string. The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- **required** boolean. Whether this param is required.
- **value** string. The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- **default** string. If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
- **allowedValues** array. The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ’Import’, ‘value’: ’import’

**arguments**  
list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

**template_script_id**  
integer optional. The ID of the template script, if any. A script cannot both have a template script and be a template for other scripts.

**notifications**  
list optional. A list containing the following elements:
- **urls** array. URLs to receive a POST request at job completion
- **successEmailSubject** string. Custom subject line for success e-mail.
- **successEmailBody** string. Custom body text for success e-mail, written in Markdown.
- **successEmailAddresses** array. Addresses to notify by e-mail when the job completes successfully.
- **successEmailFromName** string. Name from which success emails are sent; defaults to "Civis."
- **successEmailReplyTo** string. Address for replies to success emails; defaults to the author of the job.
- **failureEmailAddresses** array. Addresses to notify by e-mail when the job fails.
- **stallWarningMinutes** integer. Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

hidden boolean optional. The hidden status of the item.

Value

A list containing the following elements:

- id integer, The ID for the script.
- name string, The name of the script.
- type string, The type of the script (e.g SQL, Container, Python, R, JavaScript)
- createdAt string, The time this script was created.
- updatedAt string, The time the script was last updated.
- author list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.
- state string, The status of the script’s last run.
- finishedAt string, The time that the script’s last run finished.
- category string, The category of the script.
- projects array, An array containing the following fields:
  - id integer, The ID for the project.
  - name string, The name of the project.
- parentId integer, The ID of the parent job that will trigger this script
- userContext string, "runner" or "author", who to execute the script as when run as a template.
- params array, An array containing the following fields:
  - name string, The variable’s name as used within your code.
  - label string, The label to present to users when asking them for the value.
  - description string, A short sentence or fragment describing this parameter to the end user.
  - type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  - required boolean, Whether this param is required.
  - value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  - default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
• allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: 'label': 'Import', 'value': 'import'

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.

publishedAsTemplateId integer, The ID of the template that this script is backing.

fromTemplateId integer, The ID of the template this script uses, if any.

templateDependentsCount integer, How many other scripts use this one as a template.

templateScriptName string, The name of the template script.

links list, A list containing the following elements:

  • details string, The details link to get more information about the script.
  • runs string, The runs link to get the run information list for this script.

schedule list, A list containing the following elements:

  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:

  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  • successOn boolean, If success email notifications are on.
  • failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:

  • id integer, The ID of this user.
Description

Cancel a run

Usage

scripts_post_cancel(id)

Arguments

id integer required. The ID of the job.

Value

A list containing the following elements:

id integer, The ID of the run.
state string, The state of the run, one of 'queued', 'running' or 'cancelled'.
isCancelRequested boolean, True if run cancel requested, else false.
scripts_post_containers

Create a container

Description

Create a container

Usage

scripts_post_containers(
    required_resources,  
    docker_image_name,  
    name = NULL,  
    parent_id = NULL,  
    user_context = NULL,  
    params = NULL,  
    arguments = NULL,  
    schedule = NULL,  
    notifications = NULL,  
    repo_http_uri = NULL,  
    repo_ref = NULL,  
    remote_host_credential_id = NULL,  
    git_credential_id = NULL,  
    docker_command = NULL,  
    docker_image_tag = NULL,  
    instance_type = NULL,  
    cancel_timeout = NULL,  
    time_zone = NULL,  
    hidden = NULL,  
    target_project_id = NULL  
)

Arguments

required_resources

list required. A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares.
- memory integer, The amount of RAM to allocate for the container (in MB).
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.
- wholeInstance boolean, Whether or not to use the entire instance. If true, cpu, memory, and disk space are not required and will be set to an instance’s max.
docker_image_name
  string required. The name of the docker image to pull from DockerHub.

name
  string optional. The name of the container.

parent_id
  integer optional. The ID of the parent job that will trigger this script

user_context
  string optional. "runner" or "author", who to execute the script as when run as a template.

docker_image_name

params
  array optional. An array containing the following fields:
  • name string, The variable’s name as used within your code.
  • label string, The label to present to users when asking them for the value.
  • description string, A short sentence or fragment describing this parameter to the end user.
  • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  • required boolean, Whether this param is required.
  • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: 'Import', 'value': 'import''

arguments
  list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

schedule
  list optional. A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications
  list optional. A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
scripts_post_containers

- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

repo_http_uri string optional. The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

repo_ref string optional. The tag or branch of the github repo to clone into the container.

remote_host_credential_id integer optional. The id of the database credentials to pass into the environment of the container.

git_credential_id integer optional. The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you've submitted will be used. Unnecessary if no git repo is specified or the git repo is public.

docker_command string optional. The command to run on the container. Will be run via sh as: ['sh', '-c', dockerCommand]. Defaults to the Docker image’s ENTRY-POINT/CMD.

docker_image_tag string optional. The tag of the docker image to pull from DockerHub.

instance_type string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

cancel_timeout integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

time_zone string optional. The time zone of this script.

hidden boolean optional. The hidden status of the item.

target_project_id integer optional. Target project to which script outputs will be added.

Value

A list containing the following elements:

- id integer, The ID for the script.
- name string, The name of the container.
- type string, The type of the script (e.g Container)
- createdAt string, The time this script was created.
- updatedAt string, The time the script was last updated.
- author list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

state string, The status of the script’s last run.
finishedAt string, The time that the script’s last run finished.
category string, The category of the script.
projects array, An array containing the following fields:
  • id integer, The ID for the project.
  • name string, The name of the project.

parentId integer, The ID of the parent job that will trigger this script
userContext string, "runner" or "author", who to execute the script as when run as a template.

params array, An array containing the following fields:
  • name string, The variable’s name as used within your code.
  • label string, The label to present to users when asking them for the value.
  • description string, A short sentence or fragment describing this parameter to the end user.
  • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  • required boolean, Whether this param is required.
  • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.
templateDependentsCount integer, How many other scripts use this one as a template.
publishedAsTemplateId integer, The ID of the template that this script is backing.

fromTemplateId integer, The ID of the template script.
templateScriptName string, The name of the template script.

links list, A list containing the following elements:
  • details string, The details link to get more information about the script.
  • runs string, The runs link to get the run information list for this script.
schedule list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  • successOn boolean, If success email notifications are on.
  • failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

requiredResources list, A list containing the following elements:
  • cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares.
  • memory integer, The amount of RAM to allocate for the container (in MB).
  • diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

repoHttpUri string, The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

repoRef string, The tag or branch of the github repo to clone into the container.
remoteHostCredentialId
integer, The id of the database credentials to pass into the environment of the container.

gitCredentialId
integer, The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you’ve submitted will be used. Unnecessary if no git repo is specified or the git repo is public.

dockerCommand
string, The command to run on the container. Will be run via sh as: ["sh", ","c", dockerCommand]. Defaults to the Docker image’s ENTRYPOINT/CMD.

dockerImageName
string, The name of the docker image to pull from DockerHub.

dockerImageTag
string, The tag of the docker image to pull from DockerHub.

instanceType
string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

cancelTimeout
integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

lastRun
list, A list containing the following elements:
   • id integer,
   • state string,
   • createdAt string, The time that the run was queued.
   • startedAt string, The time that the run started.
   • finishedAt string, The time that the run completed.
   • error string, The error message for this run, if present.

timeZone
string, The time zone of this script.

hidden
boolean, The hidden status of the item.

archived
string, The archival status of the requested item(s).

targetProjectId
integer, Target project to which script outputs will be added.

scripts_post_containers_clone

Clone this Container Script

Description

Clone this Container Script

Usage

scripts_post_containers_clone(
   id,
   clone_schedule = NULL,
   clone_triggers = NULL,
   clone_notifications = NULL
)
Arguments

- **id**: integer required. The ID for the script.
- **clone_schedule**: boolean optional. If true, also copy the schedule to the new script.
- **clone_triggers**: boolean optional. If true, also copy the triggers to the new script.
- **clone_notifications**: boolean optional. If true, also copy the notifications to the new script.

Value

A list containing the following elements:

- **id**: integer, The ID for the script.
- **name**: string, The name of the container.
- **type**: string, The type of the script (e.g. Container).
- **createdAt**: string, The time this script was created.
- **updatedAt**: string, The time the script was last updated.
- **author**: list, A list containing the following elements:
  - **id**: integer, The ID of this user.
  - **name**: string, This user’s name.
  - **username**: string, This user’s username.
  - **initials**: string, This user’s initials.
  - **online**: boolean, Whether this user is online.
- **state**: string, The status of the script’s last run.
- **finishedAt**: string, The time that the script’s last run finished.
- **category**: string, The category of the script.
- **projects**: array, An array containing the following fields:
  - **id**: integer, The ID for the project.
  - **name**: string, The name of the project.
- **parentId**: integer, The ID of the parent job that will trigger this script.
- **userContext**: string, "runner" or "author", who to execute the script as when run as a template.
- **params**: array, An array containing the following fields:
  - **name**: string, The variable’s name as used within your code.
  - **label**: string, The label to present to users when asking them for the value.
  - **description**: string, A short sentence or fragment describing this parameter to the end user.
  - **type**: string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  - **required**: boolean, Whether this param is required.
  - **value**: string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
• default string. If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
• allowedValues array. The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’’

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.

templateDependentsCount integer, How many other scripts use this one as a template.

publishedAsTemplateId integer, The ID of the template that this script is backing.

fromTemplateId integer, The ID of the template script.

templateScriptName string, The name of the template script.

links list, A list containing the following elements:
• details string, The details link to get more information about the script.
• runs string, The runs link to get the run information list for this script.

schedule list, A list containing the following elements:
• scheduled boolean, If the item is scheduled.
• scheduledDays array, Day based on numeric value starting at 0 for Sunday.
• scheduledHours array, Hours of the day it is scheduled on.
• scheduledMinutes array, Minutes of the day it is scheduled on.
• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
• urls array, URLs to receive a POST request at job completion
• successEmailSubject string, Custom subject line for success e-mail.
• successEmailBody string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

requiredResources list, A list containing the following elements:
• cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares.
• memory integer, The amount of RAM to allocate for the container (in MB).
• diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

repoHttpUri string, The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

repoRef string, The tag or branch of the github repo to clone into the container.

remoteHostCredentialId integer, The id of the database credentials to pass into the environment of the container.

gitCredentialId integer, The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you’ve submitted will be used. Unnecessary if no git repo is specified or the git repo is public.

dockerCommand string, The command to run on the container. Will be run via sh as: ["sh", ",-c", dockerCommand]. Defaults to the Docker image’s ENTRYPOINT/CMD.

dockerImageName string, The name of the docker image to pull from DockerHub.

dockerImageTag string, The tag of the docker image to pull from DockerHub.

instanceType string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

cancelTimeout integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

lastRun list, A list containing the following elements:
• id integer,
• state string,
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.
• error string, The error message for this run, if present.
scripts_post_containers_runs

```plaintext
timeZone    string, The time zone of this script.
hidden      boolean, The hidden status of the item.
archived    string, The archival status of the requested item(s).
targetProjectId
             integer, Target project to which script outputs will be added.
```

---

**scripts_post_containers_runs**

*Start a run*

**Description**

Start a run

**Usage**

`scripts_post_containers_runs(id)`

**Arguments**

**id**

integer required. The ID of the container.

**Value**

A list containing the following elements:

**id**

integer, The ID of the run.

**containerId**

integer, The ID of the container.

**state**

string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

**isCancelRequested**

boolean, True if run cancel requested, else false.

**startedAt**

string, The time the last run started at.

**finishedAt**

string, The time the last run completed.

**error**

string, The error, if any, returned by the run.
scripts_post_containers_runs_logs

Add log messages

Description

Add log messages

Usage

scripts_post_containers_runs_logs(
id,  
run_id,  
message = NULL,  
level = NULL,  
messages = NULL,  
child_job_id = NULL  
)

Arguments

id           integer required. The ID of the script.
run_id       integer required. The ID of the script run.
message      string optional. The log message to store.
level        string optional. The log level of this message [default: info]
messages     array optional. An array containing the following fields:
              • message string, The log message to store.
              • level string, The log level of this message [default: info]
              • createdAt string, The timestamp of this message in ISO 8601 format. This is what logs are ordered by, so it is recommended to use timestamps with nanosecond precision. If absent, defaults to the time that the log was received by the API.
child_job_id integer optional. The ID of the child job the message came from.

Value

An empty HTTP response
scripts_post_containers_runs_outputs

Add an output for a run

Description

Add an output for a run

Usage

scripts_post_containers_runs_outputs(id, run_id, object_type, object_id)

Arguments

id
integer required. The ID of the container script.

run_id
integer required. The ID of the run.

object_type
string required. The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue

object_id
integer required. The ID of the output.

Value

A list containing the following elements:

objectType
string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue

objectId
integer, The ID of the output.

name
string, The name of the output.

link
string, The hypermedia link to the output.

value
string,

scripts_post_custom

Create a Custom Script

Description

Create a Custom Script
Usage

```python
scripts_post_custom(
    from_template_id,
    name = NULL,
    parent_id = NULL,
    arguments = NULL,
    remote_host_id = NULL,
    credential_id = NULL,
    schedule = NULL,
    notifications = NULL,
    time_zone = NULL,
    hidden = NULL,
    target_project_id = NULL
)
```

Arguments

- `from_template_id` integer required. The ID of the template script.
- `name` string optional. The name of the script.
- `parent_id` integer optional. The ID of the parent job that will trigger this script
- `arguments` list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
- `remote_host_id` integer optional. The remote host ID that this script will connect to.
- `credential_id` integer optional. The credential that this script will use.
- `schedule` list optional. A list containing the following elements:
  - `scheduled` boolean, If the item is scheduled.
  - `scheduledDays` array, Day based on numeric value starting at 0 for Sunday.
  - `scheduledHours` array, Hours of the day it is scheduled on.
  - `scheduledMinutes` array, Minutes of the day it is scheduled on.
  - `scheduledRunsPerHour` integer, Alternative to scheduled minutes, number of times to run per hour.
- `notifications` list optional. A list containing the following elements:
  - `urls` array, URLs to receive a POST request at job completion
  - `successEmailSubject` string, Custom subject line for success e-mail.
  - `successEmailBody` string, Custom body text for success e-mail, written in Markdown.
  - `successEmailAddresses` array, Addresses to notify by e-mail when the job completes successfully.
  - `successEmailFromName` string, Name from which success emails are sent; defaults to "Civis."
  - `successEmailReplyTo` string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

time_zone string optional. The time zone of this script.
hidden boolean optional. The hidden status of the item.
target_project_id integer optional. Target project to which script outputs will be added.

Value

A list containing the following elements:

id integer, The ID for the script.
name string, The name of the script.
type string, The type of the script (e.g Custom)
createdAt string, The time this script was created.
updatedAt string, The time the script was last updated.
author list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.
state string, The status of the script’s last run.
finishedAt string, The time that the script’s last run finished.
category string,
projects array, An array containing the following fields:
  - id integer, The ID for the project.
  - name string, The name of the project.
parent_id integer, The ID of the parent job that will trigger this script
params array, An array containing the following fields:
  - name string, The variable’s name as used within your code.
  - label string, The label to present to users when asking them for the value.
  - description string, A short sentence or fragment describing this parameter to the end user.
  - type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  - required boolean, Whether this param is required.
• value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
• default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
• allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.

publishedAsTemplateId integer, The ID of the template that this script is backing.

fromTemplateId integer, The ID of the template script.

uiReportUrl integer, The url of the custom HTML.

uiReportId integer, The id of the report with the custom HTML.

uiReportProvideAPIKey boolean, Whether the ui report requests an API Key from the report viewer.

templateScriptName string, The name of the template script.

templateNote string, The template’s note.

remoteHostId integer, The remote host ID that this script will connect to.

credentialId integer, The credential that this script will use.

codePreview string, The code that this script will run with arguments inserted.

schedule list, A list containing the following elements:
• scheduled boolean, If the item is scheduled.
• scheduledDays array, Day based on numeric value starting at 0 for Sunday.
• scheduledHours array, Hours of the day it is scheduled on.
• scheduledMinutes array, Minutes of the day it is scheduled on.
• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
• urls array, URLs to receive a POST request at job completion
• successEmailSubject string, Custom subject line for success e-mail.
• successEmailBody string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
scripts_post_custom_clone

Clone this Custom Script

Description

Clone this Custom Script
Usage

```python
scripts_post_custom_clone(
    id,
    clone_schedule = NULL,
    clone_triggers = NULL,
    clone_notifications = NULL
)
```

Arguments

- `id`: integer required. The ID for the script.
- `clone_schedule`: boolean optional. If true, also copy the schedule to the new script.
- `clone_triggers`: boolean optional. If true, also copy the triggers to the new script.
- `clone_notifications`: boolean optional. If true, also copy the notifications to the new script.

Value

A list containing the following elements:

- `id`: integer, The ID for the script.
- `name`: string, The name of the script.
- `type`: string, The type of the script (e.g, Custom)
- `createdAt`: string, The time this script was created.
- `updatedAt`: string, The time the script was last updated.
- `author`: list, A list containing the following elements:
  - `id`: integer, The ID of this user.
  - `name`: string, This user’s name.
  - `username`: string, This user’s username.
  - `initials`: string, This user’s initials.
  - `online`: boolean, Whether this user is online.
- `state`: string, The status of the script’s last run.
- `finishedAt`: string, The time that the script’s last run finished.
- `category`: string,
- `projects`: array, An array containing the following fields:
  - `id`: integer, The ID for the project.
  - `name`: string, The name of the project.
- `parent_id`: integer, The ID of the parent job that will trigger this script
- `params`: array, An array containing the following fields:
  - `name`: string, The variable’s name as used within your code.
  - `label`: string, The label to present to users when asking them for the value.
  - `description`: string, A short sentence or fragment describing this parameter to the end user.
scripts_post_custom_clone

- **type** string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- **required** boolean, Whether this param is required.
- **value** string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- **default** string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
- **allowedValues** array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’

**arguments** list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

**isTemplate** boolean, Whether others scripts use this one as a template.

**publishedAsTemplateId** integer, The ID of the template that this script is backing.

**fromTemplateId** integer, The ID of the template script.

**uiReportUrl** integer, The url of the custom HTML.

**uiReportId** integer, The id of the report with the custom HTML.

**uiReportProvideAPIKey** boolean, Whether the ui report requests an API Key from the report viewer.

**templateScriptName** string, The name of the template script.

**templateNote** string, The template’s note.

**remoteHostId** integer, The remote host ID that this script will connect to.

**credentialId** integer, The credential that this script will use.

**codePreview** string, The code that this script will run with arguments inserted.

**schedule** list, A list containing the following elements:
  - **scheduled** boolean, If the item is scheduled.
  - **scheduledDays** array, Day based on numeric value starting at 0 for Sunday.
  - **scheduledHours** array, Hours of the day it is scheduled on.
  - **scheduledMinutes** array, Minutes of the day it is scheduled on.
  - **scheduledRunsPerHour** integer, Alternative to scheduled minutes, number of times to run per hour.

**notifications** list, A list containing the following elements:
  - **urls** array, URLs to receive a POST request at job completion
  - **successEmailSubject** string, Custom subject line for success e-mail.
  - **successEmailBody** string, Custom body text for success e-mail, written in Markdown.
  - **successEmailAddresses** array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."

• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.

• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.

• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.

• successOn boolean, If success email notifications are on.

• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:

• id integer, The ID of this user.

• name string, This user’s name.

• username string, This user’s username.

• initials string, This user’s initials.

• online boolean, Whether this user is online.

timeZone string, The time zone of this script.

timeRun list, A list containing the following elements:

• id integer,

• state string,

• createdAt string, The time that the run was queued.

• startedAt string, The time that the run started.

• finishedAt string, The time that the run completed.

• error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.

archived string, The archival status of the requested item(s).

targetProjectId integer, Target project to which script outputs will be added.

lastSuccessfulRun list, A list containing the following elements:

• id integer,

• state string,

• createdAt string, The time that the run was queued.

• startedAt string, The time that the run started.

• finishedAt string, The time that the run completed.

• error string, The error message for this run, if present.
scripts_post_custom_runs

Start a run

Description
Start a run

Usage
scripts_post_custom_runs(id)

Arguments
id  integer required. The ID of the custom.

Value
A list containing the following elements:

id  integer, The ID of the run.
customId  integer, The ID of the custom.
state  string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested  boolean, True if run cancel requested, else false.
startedAt  string, The time the last run started at.
finishedAt  string, The time the last run completed.
error  string, The error, if any, returned by the run.

scripts_post_custom_runs_outputs

Add an output for a run

Description
Add an output for a run

Usage
scripts_post_custom_runs_outputs(id, run_id, object_type, object_id)
Arguments

- id: integer required. The ID of the custom script.
- run_id: integer required. The ID of the run.
- object_type: string required. The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
- object_id: integer required. The ID of the output.

Value

A list containing the following elements:

- objectType: string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
- objectId: integer, The ID of the output.
- name: string, The name of the output.
- link: string, The hypermedia link to the output.
- value: string,

---

_create a JavaScript Script_

**Description**

Create a JavaScript Script

**Usage**

```javascript
scripts_post_javascript(
    name,
    source,
    remote_host_id,
    credential_id,
    parent_id = NULL,
    user_context = NULL,
    params = NULL,
    arguments = NULL,
    schedule = NULL,
    notifications = NULL,
    next_run_at = NULL,
    time_zone = NULL,
    hidden = NULL,
    target_project_id = NULL
)
```
Arguments

- **name**: string required. The name of the script.
- **source**: string required. The body/text of the script.
- **remote_host_id**: integer required. The remote host ID that this script will connect to.
- **credential_id**: integer required. The credential that this script will use.
- **parent_id**: integer optional. The ID of the parent job that will trigger this script.
- **user_context**: string optional. "runner" or "author", who to execute the script as when run as a template.
- **params**: array optional. An array containing the following fields:
  - **name**: string. The variable’s name as used within your code.
  - **label**: string. The label to present to users when asking them for the value.
  - **description**: string. A short sentence or fragment describing this parameter to the end user.
  - **type**: string. The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  - **required**: boolean. Whether this param is required.
  - **value**: string. The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  - **default**: string. If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  - **allowedValues**: array. The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label': ‘Import’, ‘value': ‘import’

- **arguments**: list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

- **schedule**: list optional. A list containing the following elements:
  - **scheduled**: boolean. If the item is scheduled.
  - **scheduledDays**: array. Day based on numeric value starting at 0 for Sunday.
  - **scheduledHours**: array. Hours of the day it is scheduled on.
  - **scheduledMinutes**: array. Minutes of the day it is scheduled on.
  - **scheduledRunsPerHour**: integer. Alternative to scheduled minutes, number of times to run per hour.

- **notifications**: list optional. A list containing the following elements:
  - **urls**: array. URLs to receive a POST request at job completion
  - **successEmailSubject**: string. Custom subject line for success e-mail.
  - **successEmailBody**: string. Custom body text for success e-mail, written in Markdown.
  - **successEmailAddresses**: array. Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

next_run_at string optional. The time of the next scheduled run.
time_zone string optional. The time zone of this script.
hidden boolean optional. The hidden status of the item.
target_project_id integer optional. Target project to which script outputs will be added.

Value
A list containing the following elements:

id integer, The ID for the script.
name string, The name of the script.
type string, The type of the script (e.g SQL, Container, Python, R, JavaScript)
createdAt string, The time this script was created.
updatedAt string, The time the script was last updated.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
state string, The status of the script’s last run.
finishedAt string, The time that the script’s last run finished.
category string, The category of the script.
projects array, An array containing the following fields:
  • id integer, The ID for the project.
  • name string, The name of the project.
parentId integer, The ID of the parent job that will trigger this script
userContext string, "runner" or "author", who to execute the script as when run as a template.
params array, An array containing the following fields:
  • name string, The variable’s name as used within your code.
  • label string, The label to present to users when asking them for the value.
• description string, A short sentence or fragment describing this parameter to the end user.
• type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
• required boolean, Whether this param is required.
• value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
• default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
• allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.

publishedAsTemplateId integer, The ID of the template that this script is backing.

fromTemplateId integer, The ID of the template this script uses, if any.

templateDependentsCount integer, How many other scripts use this one as a template.

templateScriptName string, The name of the template script.

links list, A list containing the following elements:

  • details string, The details link to get more information about the script.
  • runs string, The runs link to get the run information list for this script.

schedule list, A list containing the following elements:

  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:

  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.
timeZone string, The time zone of this script.
lastRun list, A list containing the following elements:
• id integer,
• state string,
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.
• error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.
targetProjectId integer, Target project to which script outputs will be added.
archived string, The archival status of the requested item(s).
source string, The body/text of the script.
remoteHostId integer, The remote host ID that this script will connect to.
credentialId integer, The credential that this script will use.

---

Clone this JavaScript Script

Description

Clone this JavaScript Script
Usage

```python
scripts_post_javascript_clone(
    id,
    clone_schedule = NULL,
    clone_triggers = NULL,
    clone_notifications = NULL
)
```

Arguments

- **id**
  integer required. The ID for the script.
- **clone_schedule**
  boolean optional. If true, also copy the schedule to the new script.
- **clone_triggers**
  boolean optional. If true, also copy the triggers to the new script.
- **clone_notifications**
  boolean optional. If true, also copy the notifications to the new script.

Value

A list containing the following elements:

- **id**
  integer, The ID for the script.
- **name**
  string, The name of the script.
- **type**
  string, The type of the script (e.g SQL, Container, Python, R, JavaScript)
- **createdAt**
  string, The time this script was created.
- **updatedAt**
  string, The time the script was last updated.
- **author**
  list, A list containing the following elements:
    - **id**
      integer, The ID of this user.
    - **name**
      string, This user’s name.
    - **username**
      string, This user’s username.
    - **initials**
      string, This user’s initials.
    - **online**
      boolean, Whether this user is online.
- **state**
  string, The status of the script’s last run.
- **finishedAt**
  string, The time that the script’s last run finished.
- **category**
  string, The category of the script.
- **projects**
  array, An array containing the following fields:
    - **id**
      integer, The ID for the project.
    - **name**
      string, The name of the project.
- **parentId**
  integer, The ID of the parent job that will trigger this script
- **userContext**
  string, "runner" or "author", who to execute the script as when run as a template.
- **params**
  array, An array containing the following fields:
    - **name**
      string, The variable’s name as used within your code.
    - **label**
      string, The label to present to users when asking them for the value.
• description string, A short sentence or fragment describing this parameter to the end user.
• type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
• required boolean, Whether this param is required.
• value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
• default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
• allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ’Import’, ‘value’: ’import’

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.
publishedAsTemplateId integer, The ID of the template that this script is backing.
fromTemplateId integer, The ID of the template this script uses, if any.
templateDependentsCount integer, How many other scripts use this one as a template.
templateScriptName string, The name of the template script.

links list, A list containing the following elements:
  • details string, The details link to get more information about the script.
  • runs string, The runs link to get the run information list for this script.

schedule list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
scripts_post_javascript_git_commits

Commit and push a new version of the file

Description

Commit and push a new version of the file.
scripts_post_javascript_runs

Usage

```javascript
scripts_post_javascript_git_commits(id, content, message, file_hash)
```

Arguments

- **id**  
  integer required. The ID of the file.
- **content**  
  string required. The contents to commit to the file.
- **message**  
  string required. A commit message describing the changes being made.
- **file_hash**  
  string required. The full SHA of the file being replaced.

Value

A list containing the following elements:

- **content**  
  string, The file’s contents.
- **type**  
  string, The file’s type.
- **size**  
  integer, The file’s size.
- **fileHash**  
  string, The SHA of the file.

---

start a run

---

scripts_post_javascript_runs

Description

Start a run

Usage

```javascript
scripts_post_javascript_runs(id)
```

Arguments

- **id**  
  integer required. The ID of the javascript.

Value

A list containing the following elements:

- **id**  
  integer, The ID of the run.
- **javascriptId**  
  integer, The ID of the javascript.
- **state**  
  string, The state of the run, one of ‘queued’ ‘running’ ‘succeeded’ ‘failed’ or ‘cancelled’.
- **isCancelRequested**  
  boolean, True if run cancel requested, else false.
- **startedAt**  
  string, The time the last run started at.
- **finishedAt**  
  string, The time the last run completed.
- **error**  
  string, The error, if any, returned by the run.
scripts_post_javascript_runs_outputs

Add an output for a run

Description

Add an output for a run

Usage

scripts_post_javascript_runs_outputs(id, run_id, object_type, object_id)

Arguments

id
integer required. The ID of the javascript script.
run_id
integer required. The ID of the run.
object_type
string required. The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
object_id
integer required. The ID of the output.

Value

A list containing the following elements:

objectType
string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId
integer, The ID of the output.
name
string, The name of the output.
link
string, The hypermedia link to the output.
value
string,

scripts_post_python3
Create a Python Script

Description

Create a Python Script
Usage

```
scripts_post_python3(
    name,
    source,
    parent_id = NULL,
    user_context = NULL,
    params = NULL,
    arguments = NULL,
    schedule = NULL,
    notifications = NULL,
    next_run_at = NULL,
    time_zone = NULL,
    hidden = NULL,
    target_project_id = NULL,
    required_resources = NULL,
    instance_type = NULL,
    cancel_timeout = NULL,
    docker_image_tag = NULL
)
```

Arguments

- **name** (string) required. The name of the script.
- **source** (string) required. The body/text of the script.
- **parent_id** (integer) optional. The ID of the parent job that will trigger this script.
- **user_context** (string) optional. "runner" or "author", who to execute the script as when run as a template.
- **params** (array) optional. An array containing the following fields:
  - **name** (string), The variable’s name as used within your code.
  - **label** (string), The label to present to users when asking them for the value.
  - **description** (string), A short sentence or fragment describing this parameter to the end user.
  - **type** (string), The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  - **required** (boolean), Whether this param is required.
  - **value** (string), The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  - **default** (string), If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  - **allowedValues** (array), The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: `label: 'Import', 'value': 'import'`
arguments  list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

schedule  list optional. A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications  list optional. A list containing the following elements:
  - urls array, URLs to receive a POST request at job completion
  - successEmailSubject string, Custom subject line for success e-mail.
  - successEmailBody string, Custom body text for success e-mail, written in Markdown.
  - successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  - successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  - successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  - failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  - stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  - successOn boolean, If success email notifications are on.
  - failureOn boolean, If failure email notifications are on.

next_run_at  string optional. The time of the next scheduled run.

time_zone  string optional. The time zone of this script.

hidden  boolean optional. The hidden status of the item.

target_project_id  integer optional. Target project to which script outputs will be added.

required_resources  list optional. A list containing the following elements:
  - cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
  - memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
  - diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

instance_type  string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.
cancel_timeout integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

docker_image_tag string optional. The tag of the docker image to pull from DockerHub.

Value

A list containing the following elements:

id integer, The ID for the script.
name string, The name of the script.
type string, The type of the script (e.g SQL, Container, Python, R, JavaScript)
createdAt string, The time this script was created.
updatedAt string, The time the script was last updated.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user's username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
state string, The status of the script’s last run.
finishedAt string, The time that the script’s last run finished.
category string, The category of the script.
projects array, An array containing the following fields:
  • id integer, The ID for the project.
  • name string, The name of the project.
parentId integer, The ID of the parent job that will trigger this script
userContext string, "runner" or "author", who to execute the script as when run as a template.
params array, An array containing the following fields:
  • name string, The variable’s name as used within your code.
  • label string, The label to present to users when asking them for the value.
  • description string, A short sentence or fragment describing this parameter to the end user.
  • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  • required boolean, Whether this param is required.
  • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
scripts_post_python3

- default string. If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array. The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: 'Import', 'value': 'import'

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.
publishedAsTemplateId integer, The ID of the template that this script is backing.
fromTemplateId integer, The ID of the template this script uses, if any.
templateDependentsCount integer, How many other scripts use this one as a template.
templateScriptName string, The name of the template script.

links list, A list containing the following elements:
  - details string, The details link to get more information about the script.
  - runs string, The runs link to get the run information list for this script.

schedule list, A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
  - urls array, URLs to receive a POST request at job completion
  - successEmailSubject string, Custom subject line for success e-mail.
  - successEmailBody string, Custom body text for success e-mail, written in Markdown.
  - successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  - successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  - successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  - failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  - stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  - successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.
timeZone string, The time zone of this script.
lastRun list, A list containing the following elements:
  • id integer,
  • state string,
  • createdAt string, The time that the run was queued.
  • startedAt string, The time that the run started.
  • finishedAt string, The time that the run completed.
  • error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.
targetProjectId integer, Target project to which script outputs will be added.
arched string, The archival status of the requested item(s).
requiredResources list, A list containing the following elements:
  • cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
  • memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
  • diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

instanceType string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.
source string, The body/text of the script.
cancelTimeout integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.
dockerImageTag string, The tag of the docker image to pull from DockerHub.
scripts_post_python3_clone

Clone this Python Script

Description

Clone this Python Script

Usage

scripts_post_python3_clone(
  id,
  clone_schedule = NULL,
  clone_triggers = NULL,
  clone_notifications = NULL
)

Arguments

id          integer required. The ID for the script.
clone_schedule boolean optional. If true, also copy the schedule to the new script.
clone_triggers  boolean optional. If true, also copy the triggers to the new script.
clone_notifications boolean optional. If true, also copy the notifications to the new script.

Value

A list containing the following elements:

id          integer, The ID for the script.
name        string, The name of the script.
type        string, The type of the script (e.g SQL, Container, Python, R, JavaScript)
createdAt   string, The time this script was created.
updatedAt   string, The time the script was last updated.
author      list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
state       string, The status of the script’s last run.
finishedAt  string, The time that the script’s last run finished.
category    string, The category of the script.
scripts_post_python3_clone

projects array, An array containing the following fields:
- id integer, The ID for the project.
- name string, The name of the project.

parentId integer, The ID of the parent job that will trigger this script.

userContext string, "runner" or "author", who to execute the script as when run as a template.

params array, An array containing the following fields:
- name string, The variable’s name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom.
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ’Import’, ‘value’: ’import’

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.

publishedAsTemplateId integer, The ID of the template that this script is backing.

fromTemplateId integer, The ID of the template this script uses, if any.

templateDependentsCount integer, How many other scripts use this one as a template.

templateScriptName string, The name of the template script.

links list, A list containing the following elements:
- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

schedule list, A list containing the following elements:
- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.
notifications list, A list containing the following elements:
• urls array, URLs to receive a POST request at job completion
• successEmailSubject string, Custom subject line for success e-mail.
• successEmailBody string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.
timeZone string, The time zone of this script.
lastRun list, A list containing the following elements:
• id integer,
• state string,
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.
• error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.
targetProjectId integer, Target project to which script outputs will be added.
archived string, The archival status of the requested item(s).
requiredResources list, A list containing the following elements:
• cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
• memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

instanceType string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

source string, The body/text of the script.

cancelTimeout integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

dockerImageTag string, The tag of the docker image to pull from DockerHub.

---

scripts_post_python3_git_commits

*Commit and push a new version of the file*

**Description**

Commit and push a new version of the file

**Usage**

scripts_post_python3_git_commits(id, content, message, file_hash)

**Arguments**

- id integer required. The ID of the file.
- content string required. The contents to commit to the file.
- message string required. A commit message describing the changes being made.
- file_hash string required. The full SHA of the file being replaced.

**Value**

A list containing the following elements:

- content string, The file’s contents.
- type string, The file’s type.
- size integer, The file’s size.
- fileHash string, The SHA of the file.
scripts_post_python3_runs

Start a run

Description

Start a run

Usage

scripts_post_python3_runs(id)

Arguments

id	integer required. The ID of the python.

Value

A list containing the following elements:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer, The ID of the run.</td>
</tr>
<tr>
<td>pythonId</td>
<td>integer, The ID of the python.</td>
</tr>
<tr>
<td>state</td>
<td>string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed'</td>
</tr>
<tr>
<td>isCancelRequested</td>
<td>boolean, True if run cancel requested, else false.</td>
</tr>
<tr>
<td>startedAt</td>
<td>string, The time the last run started at.</td>
</tr>
<tr>
<td>finishedAt</td>
<td>string, The time the last run completed.</td>
</tr>
<tr>
<td>error</td>
<td>string, The error, if any, returned by the run.</td>
</tr>
</tbody>
</table>

scripts_post_python3_runs_outputs

Add an output for a run

Description

Add an output for a run

Usage

scripts_post_python3_runs_outputs(id, run_id, object_type, object_id)
Arguments

id         integer required. The ID of the python script.
run_id     integer required. The ID of the run.
object_type string required. The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
object_id  integer required. The ID of the output.

Value

A list containing the following elements:

objectType string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
objectId    integer, The ID of the output.
name        string, The name of the output.
link        string, The hypermedia link to the output.
value       string,

Script: scripts_post_r

Create an R Script

Description

Create an R Script

Usage

scripts_post_r(
    name,
    source,
    parent_id = NULL,
    user_context = NULL,
    params = NULL,
    arguments = NULL,
    schedule = NULL,
    notifications = NULL,
    next_run_at = NULL,
    time_zone = NULL,
    hidden = NULL,
    target_project_id = NULL,
    required_resources = NULL,
    instance_type = NULL,
    cancel_timeout = NULL,
    docker_image_tag = NULL
)

Arguments

- **name**: string required. The name of the script.
- **source**: string required. The body/text of the script.
- **parent_id**: integer optional. The ID of the parent job that will trigger this script.
- **user_context**: string optional. "runner" or "author", who to execute the script as when run as a template.
- **params**: array optional. An array containing the following fields:
  - **name**: string. The variable’s name as used within your code.
  - **label**: string. The label to present to users when asking them for the value.
  - **description**: string. A short sentence or fragment describing this parameter to the end user.
  - **type**: string. The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  - **required**: boolean. Whether this param is required.
  - **value**: string. The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  - **default**: string. If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  - **allowedValues**: array. The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: 'Import', 'value': 'import'”
- **arguments**: list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
- **schedule**: list optional. A list containing the following elements:
  - **scheduled**: boolean. If the item is scheduled.
  - **scheduledDays**: array. Day based on numeric value starting at 0 for Sunday.
  - **scheduledHours**: array. Hours of the day it is scheduled on.
  - **scheduledMinutes**: array. Minutes of the day it is scheduled on.
  - **scheduledRunsPerHour**: integer. Alternative to scheduled minutes, number of times to run per hour.
- **notifications**: list optional. A list containing the following elements:
  - **urls**: array. URLs to receive a POST request at job completion
  - **successEmailSubject**: string. Custom subject line for success e-mail.
  - **successEmailBody**: string. Custom body text for success e-mail, written in Markdown.
  - **successEmailAddresses**: array. Addresses to notify by e-mail when the job completes successfully.
  - **successEmailFromName**: string. Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string. Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array. Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer. Stall warning emails will be sent after this amount of minutes.
• successOn boolean. If success email notifications are on.
• failureOn boolean. If failure email notifications are on.

next_run_at string optional. The time of the next scheduled run.
time_zone string optional. The time zone of this script.
hidden boolean optional. The hidden status of the item.
target_project_id integer optional. Target project to which script outputs will be added.
required_resources list optional. A list containing the following elements:
  • cpu integer. The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
  • memory integer. The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
  • diskSpace number. The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.
instance_type string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.
cancel_timeout integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.
docker_image_tag string optional. The tag of the docker image to pull from DockerHub.

Value

A list containing the following elements:

id integer, The ID for the script.
name string, The name of the script.
type string, The type of the script (e.g SQL, Container, Python, R, JavaScript)
createdAt string, The time this script was created.
updatedAt string, The time the script was last updated.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string. This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

state string, The status of the script’s last run.

finishedAt string, The time that the script’s last run finished.

category string, The category of the script.

projects array, An array containing the following fields:
  • id integer, The ID for the project.
  • name string, The name of the project.

parentId integer, The ID of the parent job that will trigger this script.

userContext string, "runner" or "author", who to execute the script as when run as a template.

params array, An array containing the following fields:
  • name string, The variable’s name as used within your code.
  • label string, The label to present to users when asking them for the value.
  • description string, A short sentence or fragment describing this parameter to the end user.
  • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  • required boolean, Whether this param is required.
  • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: 'label': 'Import', 'value': 'import'

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.

publishedAsTemplateId integer, The ID of the template that this script is backing.

fromTemplateId integer, The ID of the template this script uses, if any.

templateDependentsCount integer, How many other scripts use this one as a template.

templateScriptName string, The name of the template script.

links list, A list containing the following elements:
  • details string, The details link to get more information about the script.
  • runs string, The runs link to get the run information list for this script.
schedule list, A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
  - urls array, URLs to receive a POST request at job completion
  - successEmailSubject string, Custom subject line for success e-mail.
  - successEmailBody string, Custom body text for success e-mail, written in Markdown.
  - successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  - successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  - successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  - failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  - stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  - successOn boolean, If success email notifications are on.
  - failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.

timeZone string, The time zone of this script.

lastRun list, A list containing the following elements:
  - id integer,
  - state string,
  - createdAt string, The time that the run was queued.
  - startedAt string, The time that the run started.
  - finishedAt string, The time that the run completed.
  - error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.

targetProjectId integer, Target project to which script outputs will be added.
scripts_post_run

archived
string, The archival status of the requested item(s).

requiredResources
list, A list containing the following elements:
  • cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
  • memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
  • diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

instanceType
string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

source
string, The body/text of the script.

cancelTimeout
integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

dockerImageTag
string, The tag of the docker image to pull from DockerHub.

---

scripts_post_run  Run a script

---

Description
Run a script

Usage

scripts_post_run(id)

Arguments

id integer required. The ID for the script.

Value
An empty HTTP response
Description

Clone this R Script

Usage

```r
scripts_post_r_clone(
    id,
    clone_schedule = NULL,
    clone_triggers = NULL,
    clone_notifications = NULL
)
```

Arguments

- `id` integer required. The ID for the script.
- `clone_schedule` boolean optional. If true, also copy the schedule to the new script.
- `clone_triggers` boolean optional. If true, also copy the triggers to the new script.
- `clone_notifications` boolean optional. If true, also copy the notifications to the new script.

Value

A list containing the following elements:

- `id` integer, The ID for the script.
- `name` string, The name of the script.
- `type` string, The type of the script (e.g. SQL, Container, Python, R, JavaScript)
- `createdAt` string, The time this script was created.
- `updatedAt` string, The time the script was last updated.
- `author` list, A list containing the following elements:
  - `id` integer, The ID of this user.
  - `name` string, This user's name.
  - `username` string, This user's username.
  - `initials` string, This user's initials.
  - `online` boolean, Whether this user is online.
- `state` string, The status of the script's last run.
- `finishedAt` string, The time that the script's last run finished.
- `category` string, The category of the script.
- `projects` array, An array containing the following fields:
• id integer, The ID for the project.
• name string, The name of the project.

parentId integer, The ID of the parent job that will trigger this script
userContext string, "runner" or "author", who to execute the script as when run as a template.

params array, An array containing the following fields:
• name string, The variable’s name as used within your code.
• label string, The label to present to users when asking them for the value.
• description string, A short sentence or fragment describing this parameter to the end user.
• type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
• required boolean, Whether this param is required.
• value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
• default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
• allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: 'Import', 'value': 'import'"

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.
publishedAsTemplateId integer, The ID of the template that this script is backing.

fromTemplateId integer, The ID of the template this script uses, if any.
templateDependentsCount integer, How many other scripts use this one as a template.
templateScriptName string, The name of the template script.

links list, A list containing the following elements:
• details string, The details link to get more information about the script.
• runs string, The runs link to get the run information list for this script.

schedule list, A list containing the following elements:
• scheduled boolean, If the item is scheduled.
• scheduledDays array, Day based on numeric value starting at 0 for Sunday.
• scheduledHours array, Hours of the day it is scheduled on.
• scheduledMinutes array, Minutes of the day it is scheduled on.
• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.
notifications list, A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  • successOn boolean, If success email notifications are on.
  • failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.
timeZone string, The time zone of this script.
lastRun list, A list containing the following elements:
  • id integer,
  • state string,
  • createdAt string, The time that the run was queued.
  • startedAt string, The time that the run started.
  • finishedAt string, The time that the run completed.
  • error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.
targetProjectId integer, Target project to which script outputs will be added.
archived string, The archival status of the requested item(s).
requiredResources list, A list containing the following elements:
  • cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
  • memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
scripts_post_r_git_commits

Commit and push a new version of the file

Description
Commit and push a new version of the file

Usage

scripts_post_r_git_commits(id, content, message, file_hash)

Arguments

id integer required. The ID of the file.
content string required. The contents to commit to the file.
message string required. A commit message describing the changes being made.
file_hash string required. The full SHA of the file being replaced.

Value
A list containing the following elements:

corontent string, The file’s contents.
type string, The file’s type.
size integer, The file’s size.
fileHash string, The SHA of the file.

• diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

instanceType string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

source string, The body/text of the script.

cancelTimeout integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

dockerImageTag string, The tag of the docker image to pull from DockerHub.
scripts_post_r_runs  Start a run

Description
Start a run

Usage
scripts_post_r_runs(id)

Arguments
id  integer required. The ID of the r.

Value
A list containing the following elements:

id  integer, The ID of the run.
rId  integer, The ID of the r.
state  string, The state of the run, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
isCancelRequested  boolean, True if run cancel requested, else false.
startedAt  string, The time the last run started at.
finishedAt  string, The time the last run completed.
error  string, The error, if any, returned by the run.

---

scripts_post_r_runs_outputs  Add an output for a run

Description
Add an output for a run

Usage
scripts_post_r_runs_outputs(id, run_id, object_type, object_id)
**Arguments**

- **id**
  - integer required. The ID of the r script.
- **run_id**
  - integer required. The ID of the run.
- **object_type**
  - string required. The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
- **object_id**
  - integer required. The ID of the output.

**Value**

A list containing the following elements:

- **objectType**
  - string, The type of the output. Valid values are File, Table, Report, Project, Credential, or JSONValue
- **objectId**
  - integer, The ID of the output.
- **name**
  - string, The name of the output.
- **link**
  - string, The hypermedia link to the output.
- **value**
  - string,

**scripts_post_sql**

Create a SQL script

**Description**

Create a SQL script

**Usage**

```python
scripts_post_sql(
    name,
    sql,
    remote_host_id,
    credential_id,
    parent_id = NULL,
    user_context = NULL,
    params = NULL,
    arguments = NULL,
    schedule = NULL,
    notifications = NULL,
    next_run_at = NULL,
    time_zone = NULL,
    hidden = NULL,
    target_project_id = NULL,
    csv_settings = NULL
)
```
### Arguments

- **name**
  - string required. The name of the script.

- **sql**
  - string required. The raw SQL query for the script.

- **remote_host_id**
  - integer required. The remote host ID that this script will connect to.

- **credential_id**
  - integer required. The credential that this script will use.

- **parent_id**
  - integer optional. The ID of the parent job that will trigger this script.

- **user_context**
  - string optional. “runner” or “author”, who to execute the script as when run as a template.

- **params**
  - array optional. An array containing the following fields:
    - **name**
      - string. The variable’s name as used within your code.
    - **label**
      - string. The label to present to users when asking them for the value.
    - **description**
      - string. A short sentence or fragment describing this parameter to the end user.
    - **type**
      - string. The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
    - **required**
      - boolean. Whether this param is required.
    - **value**
      - string. The value you would like to set this param to. Setting this value makes this parameter a fixed param.
    - **default**
      - string. If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
    - **allowedValues**
      - array. The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ’Import’, ’value’: ’import’

- **arguments**
  - list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

- **schedule**
  - list optional. A list containing the following elements:
    - **scheduled**
      - boolean. If the item is scheduled.
    - **scheduledDays**
      - array. Day based on numeric value starting at 0 for Sunday.
    - **scheduledHours**
      - array. Hours of the day it is scheduled on.
    - **scheduledMinutes**
      - array. Minutes of the day it is scheduled on.
    - **scheduledRunsPerHour**
      - integer. Alternative to scheduled minutes, number of times to run per hour.

- **notifications**
  - list optional. A list containing the following elements:
    - **urls**
      - array. URLs to receive a POST request at job completion.
    - **successEmailSubject**
      - string. Custom subject line for success e-mail.
    - **successEmailBody**
      - string. Custom body text for success e-mail, written in Markdown.
    - **successEmailAddresses**
      - array. Addresses to notify by e-mail when the job completes successfully.
scripts_post_sql

- **successEmailFromName** string, Name from which success emails are sent; defaults to "Civis."
- **successEmailReplyTo** string, Address for replies to success emails; defaults to the author of the job.
- **failureEmailAddresses** array, Addresses to notify by e-mail when the job fails.
- **stallWarningMinutes** integer, Stall warning emails will be sent after this amount of minutes.
- **successOn** boolean, If success email notifications are on.
- **failureOn** boolean, If failure email notifications are on.

**next_run_at** string optional. The time of the next scheduled run.

**time_zone** string optional. The time zone of this script.

**hidden** boolean optional. The hidden status of the item.

**target_project_id** integer optional. Target project to which script outputs will be added.

**csv_settings** list optional. A list containing the following elements:
- **includeHeader** boolean, Whether or not to include headers in the output data. Default: true
- **compression** string, The type of compression to use, if any, one of "none", "zip", or "gzip". Default: gzip
- **columnDelimiter** string, Which delimiter to use, one of "comma", "tab", or "pipe". Default: comma
- **unquoted** boolean, Whether or not to quote fields. Default: false
- **forceMultifile** boolean, Whether or not the csv should be split into multiple files. Default: false
- **filenamePrefix** string, A user specified filename prefix for the output file to have. Default: null
- **maxFileSize** integer, The max file size, in MB, created files will be. Only available when force_multifile is true.

**Value**

A list containing the following elements:

- **id** integer, The ID for the script.
- **name** string, The name of the script.
- **type** string, The type of the script (e.g SQL, Container, Python, R, JavaScript)
- **createdAt** string, The time this script was created.
- **updatedAt** string, The time the script was last updated.
- **author** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

state
string, The status of the script’s last run.

finishedAt
string, The time that the script’s last run finished.

category
string, The category of the script.

projects
array, An array containing the following fields:
  • id integer, The ID for the project.
  • name string, The name of the project.

parentId
integer, The ID of the parent job that will trigger this script

userContext
string, “runner” or “author”, who to execute the script as when run as a template.

params
array, An array containing the following fields:
  • name string, The variable’s name as used within your code.
  • label string, The label to present to users when asking them for the value.
  • description string, A short sentence or fragment describing this parameter to the end user.
  • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential Aws, credential_redshift, or credential_custom
  • required boolean, Whether this param is required.
  • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label': ‘Import', ‘value': 'import'

arguments
list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate
boolean, Whether others scripts use this one as a template.

publishedAsTemplateId
integer, The ID of the template that this script is backing.

fromTemplateId
integer, The ID of the template this script uses, if any.

templateDependentsCount
integer, How many other scripts use this one as a template.

templateScriptName
string, The name of the template script.

links
list, A list containing the following elements:
  • details string, The details link to get more information about the script.
  • runs string, The runs link to get the run information list for this script.

schedule
list, A list containing the following elements:
scripts_post_sql

* scheduled boolean, If the item is scheduled.
* scheduledDays array, Day based on numeric value starting at 0 for Sunday.
* scheduledHours array, Hours of the day it is scheduled on.
* scheduledMinutes array, Minutes of the day it is scheduled on.
* scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
* urls array, URLs to receive a POST request at job completion
  * successEmailSubject string, Custom subject line for success e-mail.
  * successEmailBody string, Custom body text for success e-mail, written in Markdown.
  * successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  * successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  * successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  * failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  * stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  * successOn boolean, If success email notifications are on.
  * failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
* id integer, The ID of this user.
* name string, This user’s name.
* username string, This user’s username.
* initials string, This user’s initials.
* online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.
timeZone string, The time zone of this script.

lastRun list, A list containing the following elements:
* id integer,
* state string,
* createdAt string, The time that the run was queued.
* startedAt string, The time that the run started.
* finishedAt string, The time that the run completed.
* error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.
targetProjectId integer, Target project to which script outputs will be added.
archived string, The archival status of the requested item(s).
sql      string, The raw SQL query for the script.
expandedArguments list, Expanded arguments for use in injecting into different environments.
remoteHostId integer, The remote host ID that this script will connect to.
credentialId integer, The credential that this script will use.
codePreview string, The code that this script will run with arguments inserted.
csvSettings list, A list containing the following elements:
  • includeHeader boolean, Whether or not to include headers in the output data. Default: true
  • compression string, The type of compression to use, if any, one of "none", "zip", or "gzip". Default: gzip
  • columnDelimiter string, Which delimiter to use, one of "comma", "tab", or "pipe". Default: comma
  • unquoted boolean, Whether or not to quote fields. Default: false
  • forceMultifile boolean, Whether or not the csv should be split into multiple files. Default: false
  • filenamePrefix string, A user specified filename prefix for the output file to have. Default: null
  • maxFileSize integer, The max file size, in MB, created files will be. Only available when force_multifile is true.

scripts_post_sql_clone

Clone this SQL script

Description

Clone this SQL script

Usage

scripts_post_sql_clone(
  id,
  clone_schedule = NULL,
  clone_triggers = NULL,
  clone_notifications = NULL
)

Arguments

id integer required. The ID for the script.
clone_schedule boolean optional. If true, also copy the schedule to the new script.
clone_triggers boolean optional. If true, also copy the triggers to the new script.
clone_notifications boolean optional. If true, also copy the notifications to the new script.
Value

A list containing the following elements:

- **id** integer, The ID for the script.
- **name** string, The name of the script.
- **type** string, The type of the script (e.g. SQL, Container, Python, R, JavaScript)
- **createdAt** string, The time this script was created.
- **updatedAt** string, The time the script was last updated.
- **author** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **state** string, The status of the script’s last run.
- **finishedAt** string, The time that the script’s last run finished.
- **category** string, The category of the script.
- **projects** array, An array containing the following fields:
  - **id** integer, The ID for the project.
  - **name** string, The name of the project.
- **parentId** integer, The ID of the parent job that will trigger this script
- **userContext** string, "runner" or "author", who to execute the script as when run as a template.
- **params** array, An array containing the following fields:
  - **name** string, The variable’s name as used within your code.
  - **label** string, The label to present to users when asking them for the value.
  - **description** string, A short sentence or fragment describing this parameter to the end user.
  - **type** string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  - **required** boolean, Whether this param is required.
  - **value** string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  - **default** string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  - **allowedValues** array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ’Import’, ‘value’: ’import’‘
- **arguments** list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
isTemplate boolean, Whether others scripts use this one as a template.

publishedAsTemplateId integer, The ID of the template that this script is backing.

fromTemplateId integer, The ID of the template this script uses, if any.

templateDependentsCount integer, How many other scripts use this one as a template.

templateScriptName string, The name of the template script.

links list, A list containing the following elements:
  • details string, The details link to get more information about the script.
  • runs string, The runs link to get the run information list for this script.

schedule list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  • successOn boolean, If success email notifications are on.
  • failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>timeZone</td>
<td>string, The time zone of this script.</td>
</tr>
<tr>
<td>lastRun</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• id integer,</td>
</tr>
<tr>
<td></td>
<td>• state string,</td>
</tr>
<tr>
<td></td>
<td>• createdAt string, The time that the run was queued.</td>
</tr>
<tr>
<td></td>
<td>• startedAt string, The time that the run started.</td>
</tr>
<tr>
<td></td>
<td>• finishedAt string, The time that the run completed.</td>
</tr>
<tr>
<td></td>
<td>• error string, The error message for this run, if present.</td>
</tr>
<tr>
<td>hidden</td>
<td>boolean, The hidden status of the item.</td>
</tr>
<tr>
<td>targetProjectId</td>
<td>integer, Target project to which script outputs will be added.</td>
</tr>
<tr>
<td>archived</td>
<td>string, The archival status of the requested item(s).</td>
</tr>
<tr>
<td>sql</td>
<td>string, The raw SQL query for the script.</td>
</tr>
<tr>
<td>expandedArguments</td>
<td>list, Expanded arguments for use in injecting into different environments.</td>
</tr>
<tr>
<td>remoteHostId</td>
<td>integer, The remote host ID that this script will connect to.</td>
</tr>
<tr>
<td>credentialId</td>
<td>integer, The credential that this script will use.</td>
</tr>
<tr>
<td>codePreview</td>
<td>string, The code that this script will run with arguments inserted.</td>
</tr>
<tr>
<td>csvSettings</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• includeHeader boolean, Whether or not to include headers in the output</td>
</tr>
<tr>
<td></td>
<td>data. Default: true</td>
</tr>
<tr>
<td></td>
<td>• compression string, The type of compression to use, if any, one of</td>
</tr>
<tr>
<td></td>
<td>&quot;none&quot;, &quot;zip&quot;, or &quot;gzip&quot;. Default: gzip</td>
</tr>
<tr>
<td></td>
<td>• columnDelimiter string, Which delimiter to use, one of &quot;comma&quot;, &quot;tab&quot;,</td>
</tr>
<tr>
<td></td>
<td>or &quot;pipe&quot;. Default: comma</td>
</tr>
<tr>
<td></td>
<td>• unquoted boolean, Whether or not to quote fields. Default: false</td>
</tr>
<tr>
<td></td>
<td>• forceMultifile boolean, Whether or not the csv should be split into</td>
</tr>
<tr>
<td></td>
<td>multiple files. Default: false</td>
</tr>
<tr>
<td></td>
<td>• filenamePrefix string, A user specified filename prefix for the output</td>
</tr>
<tr>
<td></td>
<td>file to have. Default: null</td>
</tr>
<tr>
<td></td>
<td>• maxFileSize integer, The max file size, in MB, created files will be.</td>
</tr>
<tr>
<td></td>
<td>Only available when force_multifile is true.</td>
</tr>
</tbody>
</table>

---

**scripts_post_sql_git_commits**

**Commit and push a new version of the file**

**Description**

Commit and push a new version of the file
Usage

scripts_post_sql_git_commits(id, content, message, file_hash)

Arguments

id          integer required. The ID of the file.
content     string required. The contents to commit to the file.
message     string required. A commit message describing the changes being made.
file_hash   string required. The full SHA of the file being replaced.

Value

A list containing the following elements:

content string, The file’s contents.
type     string, The file’s type.
size     integer, The file’s size.
fileHash string, The SHA of the file.

Description

Start a run

Usage

scripts_post_sql_runs(id)

Arguments

id          integer required. The ID of the sql.

Value

A list containing the following elements:

id          integer, The ID of this run.
sqlId       integer, The ID of this sql.
state       string, The state of this run.
isCancelRequested boolean, True if run cancel requested, else false.
startedAt   string, The time the last run started.
finishedAt  string, The time that this run finished.
scripts_put_containers

Edit a container

Description

Edit a container

Usage

```python
scripts_put_containers(
    id,
    required_resources,
    docker_image_name,
    name = NULL,
    parent_id = NULL,
    user_context = NULL,
    params = NULL,
    arguments = NULL,
    schedule = NULL,
    notifications = NULL,
    repo_http_uri = NULL,
    repo_ref = NULL,
    remote_host_credential_id = NULL,
    git_credential_id = NULL,
    docker_command = NULL,
    docker_image_tag = NULL,
    instance_type = NULL,
    cancel_timeout = NULL,
    time_zone = NULL,
    target_project_id = NULL
)
```

Arguments

- **id**: integer required. The ID for the script.
- **required_resources**: list required. A list containing the following elements:
- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares.
- memory integer, The amount of RAM to allocate for the container (in MB).
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.
- wholeInstance boolean, Whether or not to use the entire instance. If true, cpu, memory, and disk space are not required and will be set to an instance’s max.

**docker_image_name**

string required. The name of the docker image to pull from DockerHub.

**name**

string optional. The name of the container.

**parent_id**

integer optional. The ID of the parent job that will trigger this script

**user_context**

string optional. "runner" or "author", who to execute the script as when run as a template.

**params**

array optional. An array containing the following fields:

- name string, The variable’s name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’

**arguments**

list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

**schedule**

list optional. A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

**notifications**

list optional. A list containing the following elements:
scripts_put_containers

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

repo_http_uri string optional. The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

repo_ref string optional. The tag or branch of the github repo to clone into the container.

remote_host_credential_id integer optional. The id of the database credentials to pass into the environment of the container.

git_credential_id integer optional. The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you’ve submitted will be used. Unnecessary if no git repo is specified or the git repo is public.

docker_command string optional. The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]. Defaults to the Docker image’s ENTRY-POINT/CMD.

docker_image_tag string optional. The tag of the docker image to pull from DockerHub.

instance_type string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

cancel_timeout integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

time_zone string optional. The time zone of this script.

target_project_id integer optional. Target project to which script outputs will be added.

Value

A list containing the following elements:

id integer, The ID for the script.
name: string, The name of the container.
type: string, The type of the script (e.g Container)
createdAt: string, The time this script was created.
updatedAt: string, The time the script was last updated.
author: list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
state: string, The status of the script’s last run.
finishedAt: string, The time that the script’s last run finished.
category: string, The category of the script.
projects: array, An array containing the following fields:
  • id integer, The ID for the project.
  • name string, The name of the project.
parentId: integer, The ID of the parent job that will trigger this script
userContext: string, "runner" or "author", who to execute the script as when run as a template.
params: array, An array containing the following fields:
  • name string, The variable’s name as used within your code.
  • label string, The label to present to users when asking them for the value.
  • description string, A short sentence or fragment describing this parameter to the end user.
  • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  • required boolean, Whether this param is required.
  • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  • default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  • allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ’Import’, ’value’: ’import’
arguments: list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.
isTemplate: boolean, Whether others scripts use this one as a template.
templateDependentsCount: integer, How many other scripts use this one as a template.
scripts_put_containers

```

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>publishedAsTemplateId</td>
<td>integer, The ID of the template that this script is backing.</td>
</tr>
<tr>
<td>fromTemplateId</td>
<td>integer, The ID of the template script.</td>
</tr>
<tr>
<td>templateScriptName</td>
<td>string, The name of the template script.</td>
</tr>
<tr>
<td>links</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• details string, The details link to get more information about the script.</td>
</tr>
<tr>
<td></td>
<td>• runs string, The runs link to get the run information list for this script.</td>
</tr>
<tr>
<td>schedule</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• scheduled boolean, If the item is scheduled.</td>
</tr>
<tr>
<td></td>
<td>• scheduledDays array, Day based on numeric value starting at 0 for Sunday.</td>
</tr>
<tr>
<td></td>
<td>• scheduledHours array, Hours of the day it is scheduled on.</td>
</tr>
<tr>
<td></td>
<td>• scheduledMinutes array, Minutes of the day it is scheduled on.</td>
</tr>
<tr>
<td></td>
<td>• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.</td>
</tr>
<tr>
<td>notifications</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• urls array, URLs to receive a POST request at job completion</td>
</tr>
<tr>
<td></td>
<td>• successEmailSubject string, Custom subject line for success e-mail.</td>
</tr>
<tr>
<td></td>
<td>• successEmailBody string, Custom body text for success e-mail, written in Markdown.</td>
</tr>
<tr>
<td></td>
<td>• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.</td>
</tr>
<tr>
<td></td>
<td>• successEmailFromName string, Name from which success emails are sent; defaults to &quot;Civis.&quot;</td>
</tr>
<tr>
<td></td>
<td>• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.</td>
</tr>
<tr>
<td></td>
<td>• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.</td>
</tr>
<tr>
<td></td>
<td>• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.</td>
</tr>
<tr>
<td></td>
<td>• successOn boolean, If success email notifications are on.</td>
</tr>
<tr>
<td></td>
<td>• failureOn boolean, If failure email notifications are on.</td>
</tr>
<tr>
<td>runningAs</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• id integer, The ID of this user.</td>
</tr>
<tr>
<td></td>
<td>• name string, This user’s name.</td>
</tr>
<tr>
<td></td>
<td>• username string, This user’s username.</td>
</tr>
<tr>
<td></td>
<td>• initials string, This user’s initials.</td>
</tr>
<tr>
<td></td>
<td>• online boolean, Whether this user is online.</td>
</tr>
<tr>
<td>requiredResources</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares.</td>
</tr>
</tbody>
</table>
```
scripts_put_containers

• memory integer, The amount of RAM to allocate for the container (in MB).
• diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

repoHttpUri string, The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

repoRef string, The tag or branch of the github repo to clone into the container.

remoteHostCredentialId integer, The id of the database credentials to pass into the environment of the container.

gitCredentialId integer, The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you’ve submitted will be used. Unnecessary if no git repo is specified or the git repo is public.

dockerCommand string, The command to run on the container. Will be run via sh as: ["sh", ",-c", dockerCommand]. Defaults to the Docker image’s ENTRYPOINT/CMD.

dockerImageName string, The name of the docker image to pull from DockerHub.

dockerImageTag string, The tag of the docker image to pull from DockerHub.

instanceType string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

cancelTimeout integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

lastRun list, A list containing the following elements:
• id integer,
• state string,
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.
• error string, The error message for this run, if present.

timeZone string, The time zone of this script.

hidden boolean, The hidden status of the item.

archived string, The archival status of the requested item(s).

targetProjectId integer, Target project to which script outputs will be added.
scripts_put_containers_archive

Update the archive status of this object

Description

Update the archive status of this object

Usage

scripts_put_containers_archive(id, status)

Arguments

id  integer required. The ID of the object.
status  boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id  integer, The ID for the script.
name  string, The name of the container.
type  string, The type of the script (e.g Container)
createdAt  string, The time this script was created.
updatedAt  string, The time the script was last updated.
author  list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
state  string, The status of the script’s last run.
finishedAt  string, The time that the script’s last run finished.
category  string, The category of the script.
projects  array, An array containing the following fields:
  • id integer, The ID for the project.
  • name string, The name of the project.
parentId  integer, The ID of the parent job that will trigger this script
userContext  string, "runner" or "author", who to execute the script as when run as a template.
params  array, An array containing the following fields:
  • name string, The variable’s name as used within your code.
• label string, The label to present to users when asking them for the value.
• description string, A short sentence or fragment describing this parameter to the end user.
• type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
• required boolean, Whether this param is required.
• value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
• default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
• allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ’Import’, ’value’: ’import’

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.

templateDependentsCount integer, How many other scripts use this one as a template.

publishedAsTemplateId integer, The ID of the template that this script is backing.

fromTemplateId integer, The ID of the template script.

templateScriptName string, The name of the template script.

links list, A list containing the following elements:
• details string, The details link to get more information about the script.
• runs string, The runs link to get the run information list for this script.

schedule list, A list containing the following elements:
• scheduled boolean, If the item is scheduled.
• scheduledDays array, Day based on numeric value starting at 0 for Sunday.
• scheduledHours array, Hours of the day it is scheduled on.
• scheduledMinutes array, Minutes of the day it is scheduled on.
• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
• urls array, URLs to receive a POST request at job completion
• successEmailSubject string, Custom subject line for success e-mail.
• successEmailBody string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

requiredResources list, A list containing the following elements:
• cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares.
• memory integer, The amount of RAM to allocate for the container (in MB).
• diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

repoHttpUri string, The location of a github repo to clone into the container, e.g. github.com/my-user/my-repo.git.

repoRef string, The tag or branch of the github repo to clone into the container.

remoteHostCredentialId integer, The id of the database credentials to pass into the environment of the container.

gitCredentialId integer, The id of the git credential to be used when checking out the specified git repo. If not supplied, the first git credential you’ve submitted will be used. Unnecessary if no git repo is specified or the git repo is public.

dockerCommand string, The command to run on the container. Will be run via sh as: ["sh", "-c", dockerCommand]. Defaults to the Docker image’s ENTRYPOINT/CMD.

dockerImageName string, The name of the docker image to pull from DockerHub.

dockerImageTag string, The tag of the docker image to pull from DockerHub.

instanceType string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

cancelTimeout integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.
lastRun list, A list containing the following elements:
  • id integer,
  • state string,
  • createdAt string, The time that the run was queued.
  • startedAt string, The time that the run started.
  • finishedAt string, The time that the run completed.
  • error string, The error message for this run, if present.

timeZone string, The time zone of this script.
hidden boolean, The hidden status of the item.
archived string, The archival status of the requested item(s).
targetProjectId integer, Target project to which script outputs will be added.
scripts_put_containers_shares_groups

Set the permissions groups has on this object

Description
Set the permissions groups has on this object

Usage
scripts_put_containers_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)

Arguments
id  integer required. The ID of the resource that is shared.
group_ids  array required. An array of one or more group IDs.
permission_level  string required. Options are: "read", "write", or "manage".
share_email_body  string optional. Custom body text for e-mail sent on a share.
send_shared_email  boolean optional. Send email to the recipients of a share.

Value
A list containing the following elements:
readers  list, A list containing the following elements:
    • users array,
    • groups array,
writers  list, A list containing the following elements:
    • users array,
    • groups array,
owners  list, A list containing the following elements:
    • users array,
    • groups array,
totalUserShares  integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares
  integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

scripts_put_containers_shares_users
  Set the permissions users have on this object

Description
  Set the permissions users have on this object

Usage
  scripts_put_containers_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
  )

Arguments
  id             integer required. The ID of the resource that is shared.
  user_ids       array required. An array of one or more user IDs.
  permission_level
    string required. Options are: "read", "write", or "manage".
  share_email_body
    string optional. Custom body text for e-mail sent on a share.
  send_shared_email
    boolean optional. Send email to the recipients of a share.

Value
  A list containing the following elements:
    readers       list, A list containing the following elements:
      • users array,
      • groups array,
    writers       list, A list containing the following elements:
      • users array,
      • groups array,
    owners        list, A list containing the following elements:
      • users array,
• groups array,

**totalUserShares**
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

**totalGroupShares**
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**scripts_put_custom**

*Replace all attributes of this Custom Script*

---

**Description**
Replace all attributes of this Custom Script

**Usage**

```python
scripts_put_custom(
    id,
    name = NULL,
    parent_id = NULL,
    arguments = NULL,
    remote_host_id = NULL,
    credential_id = NULL,
    schedule = NULL,
    notifications = NULL,
    time_zone = NULL,
    target_project_id = NULL
)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required.</td>
<td>The ID for the script.</td>
</tr>
<tr>
<td>name</td>
<td>string optional.</td>
<td>The name of the script.</td>
</tr>
<tr>
<td>parent_id</td>
<td>integer optional.</td>
<td>The ID of the parent job that will trigger this script.</td>
</tr>
<tr>
<td>arguments</td>
<td>list optional.</td>
<td>Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.</td>
</tr>
<tr>
<td>remote_host_id</td>
<td>integer optional.</td>
<td>The remote host ID that this script will connect to.</td>
</tr>
<tr>
<td>credential_id</td>
<td>integer optional.</td>
<td>The credential that this script will use.</td>
</tr>
<tr>
<td>schedule</td>
<td>list optional.</td>
<td>A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• scheduled boolean, If the item is scheduled.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• scheduledDays array, Day based on numeric value starting at 0 for Sunday.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• scheduledHours array, Hours of the day it is scheduled on.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• scheduledMinutes array, Minutes of the day it is scheduled on.</td>
</tr>
</tbody>
</table>
- `scheduledRunsPerHour` integer, Alternative to scheduled minutes, number of times to run per hour.

`notifications` list optional. A list containing the following elements:

- `urls` array, URLs to receive a POST request at job completion
- `successEmailSubject` string, Custom subject line for success e-mail.
- `successEmailBody` string, Custom body text for success e-mail, written in Markdown.
- `successEmailAddresses` array, Addresses to notify by e-mail when the job completes successfully.
- `successEmailFromName` string, Name from which success emails are sent; defaults to "Civis."
- `successEmailReplyTo` string, Address for replies to success emails; defaults to the author of the job.
- `failureEmailAddresses` array, Addresses to notify by e-mail when the job fails.
- `stallWarningMinutes` integer, Stall warning emails will be sent after this amount of minutes.
- `successOn` boolean, If success email notifications are on.
- `failureOn` boolean, If failure email notifications are on.

`time_zone` string optional. The time zone of this script.

`target_project_id` integer optional. Target project to which script outputs will be added.

**Value**

A list containing the following elements:

- `id` integer, The ID for the script.
- `name` string, The name of the script.
- `type` string, The type of the script (e.g Custom)
- `createdAt` string, The time this script was created.
- `updatedAt` string, The time the script was last updated.
- `author` list, A list containing the following elements:
  - `id` integer, The ID of this user.
  - `name` string, This user’s name.
  - `username` string, This user’s username.
  - `initials` string, This user’s initials.
  - `online` boolean, Whether this user is online.

- `state` string, The status of the script’s last run.
- `finishedAt` string, The time that the script’s last run finished.
- `category` string,
- `projects` array, An array containing the following fields:
- id integer, The ID for the project.
- name string, The name of the project.

parentId integer, The ID of the parent job that will trigger this script

params array, An array containing the following fields:
- name string, The variable’s name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.

publishedAsTemplateId integer, The ID of the template that this script is backing.

fromTemplateId integer, The ID of the template script.

uiReportUrl integer, The url of the custom HTML.

uiReportId integer, The id of the report with the custom HTML.

uiReportProvideAPIKey boolean, Whether the ui report requests an API Key from the report viewer.

templateScriptName string, The name of the template script.

templateNote string, The template’s note.

remoteHostId integer, The remote host ID that this script will connect to.

credentialId integer, The credential that this script will use.

codePreview string, The code that this script will run with arguments inserted.

schedule list, A list containing the following elements:
- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  • successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  • successOn boolean, If success email notifications are on.
  • failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user's name.
  • username string, This user's username.
  • initials string, This user's initials.
  • online boolean, Whether this user is online.

timeZone string, The time zone of this script.

lastRun list, A list containing the following elements:
  • id integer,
  • state string,
  • createdAt string, The time that the run was queued.
  • startedAt string, The time that the run started.
  • finishedAt string, The time that the run completed.
  • error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.

archived string, The archival status of the requested item(s).

targetProjectId integer, Target project to which script outputs will be added.

lastSuccessfulRun list, A list containing the following elements:
  • id integer,
  • state string,
  • createdAt string, The time that the run was queued.
scripts_put_custom_archive

Update the archive status of this object

Description

Update the archive status of this object

Usage

scripts_put_custom_archive(id, status)

Arguments

id integer required. The ID of the object.
status boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id integer, The ID for the script.
name string, The name of the script.
type string, The type of the script (e.g Custom)
createdAt string, The time this script was created.
updatedAt string, The time the script was last updated.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
state string, The status of the script’s last run.
finishedAt string, The time that the script’s last run finished.
category string,
projects array, An array containing the following fields:
  • id integer, The ID for the project.
  • name string, The name of the project.
parentId integer, The ID of the parent job that will trigger this script
params array, An array containing the following fields:

• name string, The variable’s name as used within your code.
• label string, The label to present to users when asking them for the value.
• description string, A short sentence or fragment describing this parameter to the end user.
• type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
• required boolean, Whether this param is required.
• value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
• default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
• allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.
publishedAsTemplateId integer, The ID of the template that this script is backing.
fromTemplateId integer, The ID of the template script.
uiReportUrl integer, The url of the custom HTML.
uiReportId integer, The id of the report with the custom HTML.
nuiReportProvideAPIKey boolean, Whether the ui report requests an API Key from the report viewer.
templateScriptName string, The name of the template script.
templateNote string, The template’s note.
remoteHostId integer, The remote host ID that this script will connect to.
credentialId integer, The credential that this script will use.
codePreview string, The code that this script will run with arguments inserted.
schedule list, A list containing the following elements:

• scheduled boolean, If the item is scheduled.
• scheduledDays array, Day based on numeric value starting at 0 for Sunday.
• scheduledHours array, Hours of the day it is scheduled on.
• scheduledMinutes array, Minutes of the day it is scheduled on.
• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
• urls array, URLs to receive a POST request at job completion
• successEmailSubject string, Custom subject line for success e-mail.
• successEmailBody string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

timeZone string, The time zone of this script.

lastRun list, A list containing the following elements:
• id integer,
• state string,
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.
• error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.

archived string, The archival status of the requested item(s).

targetProjectId integer, Target project to which script outputs will be added.

cacheRun list, A list containing the following elements:
• id integer,
• state string,
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.
• error string, The error message for this run, if present.
scripts_put_custom_projects

*Add a Custom Script to a project*

**Description**

Add a Custom Script to a project

**Usage**

```python
scripts_put_custom_projects(id, project_id)
```

**Arguments**

- **id** : integer required. The ID of the Custom Script.
- **project_id** : integer required. The ID of the project.

**Value**

An empty HTTP response

---

scripts_put_custom_shares_groups

*Set the permissions groups has on this object*

**Description**

Set the permissions groups has on this object

**Usage**

```python
scripts_put_custom_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```
scripts_put_custom_shares_users

Arguments

id  integer required. The ID of the resource that is shared.
group_ids  array required. An array of one or more group IDs.
permission_level  string required. Options are: "read", "write", or "manage".
share_email_body  string optional. Custom body text for e-mail sent on a share.
send_shared_email  boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers  list, A list containing the following elements:
  • users array,
  • groups array,

writers  list, A list containing the following elements:
  • users array,
  • groups array,

owners  list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares  integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares  integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

Description

Set the permissions users have on this object

Usage

scripts_put_custom_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)

Arguments

- **id** integer required. The ID of the resource that is shared.
- **user_ids** array required. An array of one or more user IDs.
- **permission_level** string required. Options are: "read", "write", or "manage".
- **share_email_body** string optional. Custom body text for e-mail sent on a share.
- **send_shared_email** boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,
- **writers** list, A list containing the following elements:
  - users array,
  - groups array,
- **owners** list, A list containing the following elements:
  - users array,
  - groups array,
- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**scripts_put_javascript**

Replace all attributes of this JavaScript Script

---

**Description**

Replace all attributes of this JavaScript Script
Usage

scripts_put_javascript(id, name, source, remote_host_id, credential_id, parent_id = NULL, user_context = NULL, params = NULL, arguments = NULL, schedule = NULL, notifications = NULL, next_run_at = NULL, time_zone = NULL, target_project_id = NULL)

Arguments

id integer required. The ID for the script.
name string required. The name of the script.
source string required. The body/text of the script.
remote_host_id integer required. The remote host ID that this script will connect to.
credential_id integer required. The credential that this script will use.
parent_id integer optional. The ID of the parent job that will trigger this script.
user_context string optional. "runner" or "author", who to execute the script as when run as a template.
params array optional. An array containing the following fields:
- name string. The variable’s name as used within your code.
- label string. The label to present to users when asking them for the value.
- description string. A short sentence or fragment describing this parameter to the end user.
- type string. The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean. Whether this param is required.
- value string. The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string. If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
- allowedValues array. The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ’Import’, ‘value’: ’import’
arguments list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

schedule list optional. A list containing the following elements:
- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list optional. A list containing the following elements:
- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

next_run_at string optional. The time of the next scheduled run.

time_zone string optional. The time zone of this script.

target_project_id integer optional. Target project to which script outputs will be added.

Value

A list containing the following elements:

id integer, The ID for the script.

name string, The name of the script.

type string, The type of the script (e.g SQL, Container, Python, R, JavaScript)

createdAt string, The time this script was created.

updatedAt string, The time the script was last updated.

author list, A list containing the following elements:
- id integer, The ID of this user.
- name string, This user’s name.
- **username** string, This user's username.
- **initials** string, This user's initials.
- **online** boolean, Whether this user is online.

**state** string, The status of the script's last run.

**finishedAt** string, The time that the script's last run finished.

**category** string, The category of the script.

**projects** array, An array containing the following fields:
  - **id** integer, The ID for the project.
  - **name** string, The name of the project.

**parentId** integer, The ID of the parent job that will trigger this script

**userContext** string, "runner" or "author", who to execute the script as when run as a template.

**params** array, An array containing the following fields:
  - **name** string, The variable’s name as used within your code.
  - **label** string, The label to present to users when asking them for the value.
  - **description** string, A short sentence or fragment describing this parameter to the end user.
  - **type** string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  - **required** boolean, Whether this param is required.
  - **value** string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  - **default** string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  - **allowedValues** array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label’: ‘Import’, ‘value’: ‘import’

**arguments** list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

**isTemplate** boolean, Whether others scripts use this one as a template.

**publishedAsTemplateId** integer, The ID of the template that this script is backing.

**fromTemplateId** integer, The ID of the template this script uses, if any.

**templateDependentsCount** integer, How many other scripts use this one as a template.

**templateScriptName** string, The name of the template script.

**links** list, A list containing the following elements:
  - **details** string, The details link to get more information about the script.
  - **runs** string, The runs link to get the run information list for this script.
schedule list, A list containing the following elements:
- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to “Civis.”
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
- id integer, The ID of this user.
- name string, This user’s name.
- username string, This user’s username.
- initials string, This user’s initials.
- online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.

timeZone string, The time zone of this script.

lastRun list, A list containing the following elements:
- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.

targetProjectId integer, Target project to which script outputs will be added.
scripts_put_javascript_archive

archived  string, The archival status of the requested item(s).
source    string, The body/text of the script.
remoteHostId  integer, The remote host ID that this script will connect to.
credentialId integer, The credential that this script will use.

Description

Update the archive status of this object

Usage

scripts_put_javascript_archive(id, status)

Arguments

id     integer required. The ID of the object.
status boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id             integer, The ID for the script.
name           string, The name of the script.
type           string, The type of the script (e.g SQL, Container, Python, R, JavaScript)
createdAt      string, The time this script was created.
updatedAt      string, The time the script was last updated.
author         list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
state          string, The status of the script’s last run.
finishedAt     string, The time that the script’s last run finished.
category       string, The category of the script.
projects       array, An array containing the following fields:
  • id integer, The ID for the project.
scripts_put_javascript_archive

- name string, The name of the project.

parentId integer, The ID of the parent job that will trigger this script

userContext string, "runner" or "author", who to execute the script as when run as a template.

params array, An array containing the following fields:
- name string, The variable’s name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: 'label': 'Import', 'value': 'import'

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.

publishedAsTemplateId integer, The ID of the template that this script is backing.

fromTemplateId integer, The ID of the template this script uses, if any.

templateDependentsCount integer, How many other scripts use this one as a template.

templateScriptName string, The name of the template script.

links list, A list containing the following elements:
- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

schedule list, A list containing the following elements:
- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
• urls array, URLs to receive a POST request at job completion
• successEmailSubject string, Custom subject line for success e-mail.
• successEmailBody string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user's name.
• username string, This user's username.
• initials string, This user's initials.
• online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.
timeZone string, The time zone of this script.
lastRun list, A list containing the following elements:
• id integer,
• state string,
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.
• error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.
targetProjectId integer, Target project to which script outputs will be added.
archived string, The archival status of the requested item(s).
source string, The body/text of the script.
remoteHostId integer, The remote host ID that this script will connect to.
credentialId integer, The credential that this script will use.
scripts_put_javascript_git

Attach an item to a file in a git repo

Description

Attach an item to a file in a git repo

Usage

scripts_put_javascript_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  pull_from_git = NULL
)

Arguments

id integer required. The ID of the file.
git_ref string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
git_branch string optional. The git branch that the file is on.
git_path string optional. The path of the file in the repository.
git_repo_url string optional. The URL of the git repository.
pull_from_git boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

gitRef string, A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
gitBranch string, The git branch that the file is on.
gitPath string, The path of the file in the repository.
gitRepo list, A list containing the following elements:
  - id integer, The ID for this git repository.
  - repoUrl string, The URL for this git repository.
  - createdAt string,
  - updatedAt string,
pullFromGit boolean, Automatically pull latest commit from git. Only works for scripts.
scripts_put_javascript_projects

*Add a JavaScript Script to a project*

**Description**

Add a JavaScript Script to a project

**Usage**

`scripts_put_javascript_projects(id, project_id)`

**Arguments**

- **id**: integer required. The ID of the JavaScript Script.
- **project_id**: integer required. The ID of the project.

**Value**

An empty HTTP response

---

scripts_put_javascript_shares_groups

*Set the permissions groups has on this object*

**Description**

Set the permissions groups has on this object

**Usage**

```python
scripts_put_javascript_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```
scripts_put_javascript_shares_users

Arguments

id integer required. The ID of the resource that is shared.
group_ids array required. An array of one or more group IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers list, A list containing the following elements:
  • users array,
  • groups array,

writers list, A list containing the following elements:
  • users array,
  • groups array,

owners list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

Description

Set the permissions users have on this object

Usage

scripts_put_javascript_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
Arguments

id
integer required. The ID of the resource that is shared.

user_ids
array required. An array of one or more user IDs.

permission_level
string required. Options are: "read", "write", or "manage".

share_email_body
string optional. Custom body text for e-mail sent on a share.

send_shared_email
boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers
list, A list containing the following elements:
• users array,
• groups array,

writers
list, A list containing the following elements:
• users array,
• groups array,

owners
list, A list containing the following elements:
• users array,
• groups array,

totalUserShares
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

Description

Replace all attributes of this Python Script
Usage

```python
scripts_put_python3(
    id,
    name,
    source,
    parent_id = NULL,
    user_context = NULL,
    params = NULL,
    arguments = NULL,
    schedule = NULL,
    notifications = NULL,
    next_run_at = NULL,
    time_zone = NULL,
    target_project_id = NULL,
    required_resources = NULL,
    instance_type = NULL,
    cancel_timeout = NULL,
    docker_image_tag = NULL
)
```

Arguments

- **id**: integer required. The ID for the script.
- **name**: string required. The name of the script.
- **source**: string required. The body/text of the script.
- **parent_id**: integer optional. The ID of the parent job that will trigger this script.
- **user_context**: string optional. "runner" or "author", who to execute the script as when run as a template.
- **params**: array optional. An array containing the following fields:
  - **name**: string. The variable’s name as used within your code.
  - **label**: string. The label to present to users when asking them for the value.
  - **description**: string. A short sentence or fragment describing this parameter to the end user.
  - **type**: string. The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  - **required**: boolean, Whether this param is required.
  - **value**: string. The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  - **default**: string. If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
  - **allowedValues**: array. The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’
arguments list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

schedule list optional. A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

next_run_at string optional. The time of the next scheduled run.

time_zone string optional. The time zone of this script.

target_project_id integer optional. Target project to which script outputs will be added.

required_resources list optional. A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

instance_type string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.
cancel_timeout integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

docker_image_tag

string optional. The tag of the docker image to pull from DockerHub.

Value

A list containing the following elements:

id integer, The ID for the script.
name string, The name of the script.
type string, The type of the script (e.g SQL, Container, Python, R, JavaScript)
createdAt string, The time this script was created.
updatedAt string, The time the script was last updated.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
state string, The status of the script’s last run.
finishedAt string, The time that the script’s last run finished.
category string, The category of the script.
projects array, An array containing the following fields:
  • id integer, The ID for the project.
  • name string, The name of the project.
parentId integer, The ID of the parent job that will trigger this script
userContext string, "runner" or "author", who to execute the script as when run as a template.
params array, An array containing the following fields:
  • name string, The variable’s name as used within your code.
  • label string, The label to present to users when asking them for the value.
  • description string, A short sentence or fragment describing this parameter to the end user.
  • type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  • required boolean, Whether this param is required.
  • value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string. If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
- allowedValues array. The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: 'label: 'Import', 'value': 'import''

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.

publishedAsTemplateId integer, The ID of the template that this script is backing.

fromTemplateId integer, The ID of the template this script uses, if any.

templateDependentsCount integer, How many other scripts use this one as a template.

templateScriptName string, The name of the template script.

links list, A list containing the following elements:
- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

schedule list, A list containing the following elements:
- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.

timeZone string, The time zone of this script.

lastRun list, A list containing the following elements:
  - id integer,
  - state string,
  - createdAt string, The time that the run was queued.
  - startedAt string, The time that the run started.
  - finishedAt string, The time that the run completed.
  - error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.

targetProjectId integer, Target project to which script outputs will be added.

archived string, The archival status of the requested item(s).

requiredResources list, A list containing the following elements:
  - cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
  - memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
  - diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

instanceType string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

source string, The body/text of the script.

cancelTimeout integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

dockerImageTag string, The tag of the docker image to pull from DockerHub.
Description

Update the archive status of this object

Usage

scripts_put_python3_archive(id, status)

Arguments

id  integer required. The ID of the object.
status  boolean required. The desired archived status of the object.

Value

A list containing the following elements:

id  integer, The ID for the script.
name  string, The name of the script.
type  string, The type of the script (e.g SQL, Container, Python, R, JavaScript)
createdAt  string, The time this script was created.
updatedAt  string, The time the script was last updated.
author  list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
state  string, The status of the script’s last run.
finishedAt  string, The time that the script’s last run finished.
category  string, The category of the script.
projects  array, An array containing the following fields:
  • id integer, The ID for the project.
  • name string, The name of the project.
parentId  integer, The ID of the parent job that will trigger this script
userContext  string, "runner" or "author", who to execute the script as when run as a template.
params  array, An array containing the following fields:
  • name string, The variable’s name as used within your code.
• label string. The label to present to users when asking them for the value.
• description string. A short sentence or fragment describing this parameter to the end user.
• type string. The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
• required boolean. Whether this param is required.
• value string. The value you would like to set this param to. Setting this value makes this parameter a fixed param.
• default string. If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
• allowedValues array. The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label’:'Import', 'value': 'import'

arguments list. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean. Whether others scripts use this one as a template.

publishedAsTemplateId integer. The ID of the template that this script is backing.

fromTemplateId integer. The ID of the template this script uses, if any.

templateDependentsCount integer. How many other scripts use this one as a template.

templateScriptName string. The name of the template script.

links list. A list containing the following elements:
• details string. The details link to get more information about the script.
• runs string. The runs link to get the run information list for this script.

schedule list. A list containing the following elements:
• scheduled boolean. If the item is scheduled.
• scheduledDays array. Day based on numeric value starting at 0 for Sunday.
• scheduledHours array. Hours of the day it is scheduled on.
• scheduledMinutes array. Minutes of the day it is scheduled on.
• scheduledRunsPerHour integer. Alternative to scheduled minutes, number of times to run per hour.

notifications list. A list containing the following elements:
• urls array. URLs to receive a POST request at job completion
• successEmailSubject string. Custom subject line for success e-mail.
• successEmailBody string. Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array. Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.

timeZone string, The time zone of this script.

lastRun list, A list containing the following elements:
  - id integer,
  - state string,
  - createdAt string, The time that the run was queued.
  - startedAt string, The time that the run started.
  - finishedAt string, The time that the run completed.
  - error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.

targetProjectId integer, Target project to which script outputs will be added.

archived string, The archival status of the requested item(s).

requiredResources list, A list containing the following elements:
  - cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
  - memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
  - diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

instanceType string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

source string, The body/text of the script.
scripts_put_python3_git

Description

Attach an item to a file in a git repo

Usage

scripts_put_python3_git(
    id,
    git_ref = NULL,
    git_branch = NULL,
    git_path = NULL,
    git_repo_url = NULL,
    pull_from_git = NULL
)

Arguments

id integer required. The ID of the file.
git_ref string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
git_branch string optional. The git branch that the file is on.
git_path string optional. The path of the file in the repository.
git_repo_url string optional. The URL of the git repository.
pull_from_git boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

gitRef string, A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
gitBranch string, The git branch that the file is on.
gitPath string, The path of the file in the repository.
gitRepo list, A list containing the following elements:
    • id integer, The ID for this git repository.
scripts_put_python3_projects

Add a Python Script to a project

Description

Add a Python Script to a project

Usage

scripts_put_python3_projects(id, project_id)

Arguments

id integer required. The ID of the Python Script.
project_id integer required. The ID of the project.

Value

An empty HTTP response

scripts_put_python3_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

scripts_put_python3_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
Arguments

id          integer required. The ID of the resource that is shared.
group_ids   array required. An array of one or more group IDs.
permission_level   string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers    list, A list containing the following elements:
• users array,
• groups array,

writers    list, A list containing the following elements:
• users array,
• groups array,

owners    list, A list containing the following elements:
• users array,
• groups array,

totalUserShares  integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

Description

Set the permissions users have on this object

Usage

scripts_put_python3_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
Arguments

id: integer required. The ID of the resource that is shared.
user_ids: array required. An array of one or more user IDs.
permission_level: string required. Options are: "read", "write", or "manage".
share_email_body: string optional. Custom body text for e-mail sent on a share.
send_shared_email: boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers: list, A list containing the following elements:
  • users array,
  • groups array,

writers: list, A list containing the following elements:
  • users array,
  • groups array,

owners: list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares: integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

scripts_put_r

Replace all attributes of this R Script

Description

Replace all attributes of this R Script
Usage

```
scripts_put_r(
    id,
    name,
    source,
    parent_id = NULL,
    user_context = NULL,
    params = NULL,
    arguments = NULL,
    schedule = NULL,
    notifications = NULL,
    next_run_at = NULL,
    time_zone = NULL,
    target_project_id = NULL,
    required_resources = NULL,
    instance_type = NULL,
    cancel_timeout = NULL,
    docker_image_tag = NULL
)
```

Arguments

- **id** integer required. The ID for the script.
- **name** string required. The name of the script.
- **source** string required. The body/text of the script.
- **parent_id** integer optional. The ID of the parent job that will trigger this script.
- **user_context** string optional. "runner" or "author", who to execute the script as when run as a template.
- **params** array optional. An array containing the following fields:
  - **name** string, The variable's name as used within your code.
  - **label** string, The label to present to users when asking them for the value.
  - **description** string, A short sentence or fragment describing this parameter to the end user.
  - **type** string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
  - **required** boolean, Whether this param is required.
  - **value** string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
  - **default** string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool's or false, False, f, n, no, or 0 for false bool's. Cannot be used for parameters that are required or a credential type.
  - **allowedValues** array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: `{'label': 'Import', 'value': 'import'}`
scripts_put_r

arguments list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

schedule list optional. A list containing the following elements:

- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list optional. A list containing the following elements:

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
- successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

next_run_at string optional. The time of the next scheduled run.

time_zone string optional. The time zone of this script.

target_project_id integer optional. Target project to which script outputs will be added.

required_resources list optional. A list containing the following elements:

- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

instance_type string optional. The EC2 instance type to deploy to. Only available for jobs running on kubernetes.
cancel_timeout integer optional. The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

docker_image_tag string optional. The tag of the docker image to pull from DockerHub.

Value
A list containing the following elements:

id integer, The ID for the script.

name string, The name of the script.

type string, The type of the script (e.g SQL, Container, Python, R, JavaScript)

createdAt string, The time this script was created.

updatedAt string, The time the script was last updated.

author list, A list containing the following elements:

• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user's username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

state string, The status of the script’s last run.

finishedAt string, The time that the script’s last run finished.

category string, The category of the script.

projects array, An array containing the following fields:

• id integer, The ID for the project.
• name string, The name of the project.

parentId integer, The ID of the parent job that will trigger this script

userContext string, "runner" or "author", who to execute the script as when run as a template.

params array, An array containing the following fields:

• name string, The variable's name as used within your code.
• label string, The label to present to users when asking them for the value.
• description string, A short sentence or fragment describing this parameter to the end user.
• type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
• required boolean, Whether this param is required.
• value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
scripts_put_r

- default string. If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
- allowedValues array. The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: 'Import', 'value': 'import''

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.
publishedAsTemplateId integer, The ID of the template that this script is backing.
fromTemplateId integer, The ID of the template this script uses, if any.
templateDependentsCount integer, How many other scripts use this one as a template.
templateScriptName string, The name of the template script.

links list, A list containing the following elements:
  - details string, The details link to get more information about the script.
  - runs string, The runs link to get the run information list for this script.

schedule list, A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
  - urls array, URLs to receive a POST request at job completion
  - successEmailSubject string, Custom subject line for success e-mail.
  - successEmailBody string, Custom body text for success e-mail, written in Markdown.
  - successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  - successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
  - successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
  - failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  - stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  - successOn boolean, If success email notifications are on.
- failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
- id integer, The ID of this user.
- name string, This user's name.
- username string, This user's username.
- initials string, This user's initials.
- online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.

timeZone string, The time zone of this script.

lastRun list, A list containing the following elements:
- id integer,
- state string,
- createdAt string, The time that the run was queued.
- startedAt string, The time that the run started.
- finishedAt string, The time that the run completed.
- error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.

targetProjectId integer, Target project to which script outputs will be added.

archived string, The archival status of the requested item(s).

requiredResources list, A list containing the following elements:
- cpu integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
- memory integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
- diskSpace number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

instanceType string, The EC2 instance type to deploy to. Only available for jobs running on kubernetes.

source string, The body/text of the script.

cancelTimeout integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

dockerImageTag string, The tag of the docker image to pull from DockerHub.
update the archive status of this object

Usage

scripts_put_r_archive(id, status)

Arguments

- **id**: integer required. The ID of the object.
- **status**: boolean required. The desired archived status of the object.

Value

A list containing the following elements:

- **id**: integer, The ID for the script.
- **name**: string, The name of the script.
- **type**: string, The type of the script (e.g. SQL, Container, Python, R, JavaScript)
- **createdAt**: string, The time this script was created.
- **updatedAt**: string, The time the script was last updated.
- **author**: list, A list containing the following elements:
  - **id**: integer, The ID of this user.
  - **name**: string, This user’s name.
  - **username**: string, This user’s username.
  - **initials**: string, This user’s initials.
  - **online**: boolean, Whether this user is online.
- **state**: string, The status of the script’s last run.
- **finishedAt**: string, The time that the script’s last run finished.
- **category**: string, The category of the script.
- **projects**: array, An array containing the following fields:
  - **id**: integer, The ID for the project.
  - **name**: string, The name of the project.
- **parentJobId**: integer, The ID of the parent job that will trigger this script
- **userContext**: string, "runner" or "author", who to execute the script as when run as a template.
- **params**: array, An array containing the following fields:
  - **name**: string, The variable’s name as used within your code.
  - **label**: string, The label to present to users when asking them for the value.
scripts_put_r_archive

- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’

arguments
list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate
boolean, Whether others scripts use this one as a template.

publishedAsTemplateId
integer, The ID of the template that this script is backing.

fromTemplateId
integer, The ID of the template this script uses, if any.

templateDependentsCount
integer, How many other scripts use this one as a template.

templateScriptName
string, The name of the template script.

links
list, A list containing the following elements:
- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

schedule
list, A list containing the following elements:
- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications
list, A list containing the following elements:
- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• `successEmailFromName` string, Name from which success emails are sent; defaults to "Civis."
• `successEmailReplyTo` string, Address for replies to success emails; defaults to the author of the job.
• `failureEmailAddresses` array, Addresses to notify by e-mail when the job fails.
• `stallWarningMinutes` integer, Stall warning emails will be sent after this amount of minutes.
• `successOn` boolean, If success email notifications are on.
• `failureOn` boolean, If failure email notifications are on.

`runningAs` list, A list containing the following elements:
• `id` integer, The ID of this user.
• `name` string, This user’s name.
• `username` string, This user’s username.
• `initials` string, This user’s initials.
• `online` boolean, Whether this user is online.

`nextRunAt` string, The time of the next scheduled run.

`timeZone` string, The time zone of this script.

`lastRun` list, A list containing the following elements:
• `id` integer,
• `state` string,
• `createdAt` string, The time that the run was queued.
• `startedAt` string, The time that the run started.
• `finishedAt` string, The time that the run completed.
• `error` string, The error message for this run, if present.

`hidden` boolean, The hidden status of the item.

`targetProjectId` integer, Target project to which script outputs will be added.

`archived` string, The archival status of the requested item(s).

`requiredResources` list, A list containing the following elements:
• `cpu` integer, The number of CPU shares to allocate for the container. Each core has 1000 shares. Must be at least 2 shares.
• `memory` integer, The amount of RAM to allocate for the container (in MB). Must be at least 4 MB.
• `diskSpace` number, The amount of disk space, in GB, to allocate for the container. This space will be used to hold the git repo configured for the container and anything your container writes to /tmp or /data. Fractional values (e.g. 0.25) are supported.

`instanceType` string, The EC2 instance type to deploy to. Only available for jobs running on Kubernetes.

`source` string, The body/text of the script.
cancelTimeout integer, The amount of time (in seconds) to wait before forcibly terminating the script. When the script is cancelled, it is first sent a TERM signal. If the script is still running after the timeout, it is sent a KILL signal. Defaults to 0.

dockerImageTag string, The tag of the docker image to pull from DockerHub.

---

scripts_put_r_git Attach an item to a file in a git repo

Description

Attach an item to a file in a git repo

Usage

```r
scripts_put_r_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  pull_from_git = NULL
)
```

Arguments

- `id` integer required. The ID of the file.
- `git_ref` string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
- `git_branch` string optional. The git branch that the file is on.
- `git_path` string optional. The path of the file in the repository.
- `git_repo_url` string optional. The URL of the git repository.
- `pull_from_git` boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

- `gitRef` string, A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
- `gitBranch` string, The git branch that the file is on.
- `gitPath` string, The path of the file in the repository.
- `gitRepo` list, A list containing the following elements:
  - `id` integer, The ID for this git repository.
• repoUrl string. The URL for this git repository.
• createdAt string.
• updatedAt string.

pullFromGit     boolean, Automatically pull latest commit from git. Only works for scripts.

---

**scripts_put_r_projects**

*Add an R Script to a project*

**Description**

Add an R Script to a project

**Usage**

```r
scripts_put_r_projects(id, project_id)
```

**Arguments**

- `id` integer required. The ID of the R Script.
- `project_id` integer required. The ID of the project.

**Value**

An empty HTTP response

---

**scripts_put_r_shares_groups**

*Set the permissions groups has on this object*

**Description**

Set the permissions groups has on this object

**Usage**

```r
scripts_put_r_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```
scripts_put_r_shares_users

Description
Set the permissions users have on this object

Usage
scripts_put_r_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
Arguments

id integer required. The ID of the resource that is shared.
user_ids array required. An array of one or more user IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers list, A list containing the following elements:
  • users array,
  • groups array,

writers list, A list containing the following elements:
  • users array,
  • groups array,

owners list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

Description

Replace all attributes of this SQL script
Usage

scripts_put_sql(
    id,
    name,
    sql,
    remote_host_id,
    credential_id,
    parent_id = NULL,
    user_context = NULL,
    params = NULL,
    arguments = NULL,
    schedule = NULL,
    notifications = NULL,
    next_run_at = NULL,
    time_zone = NULL,
    target_project_id = NULL,
    csv_settings = NULL
)

Arguments

id integer required. The ID for the script.
name string required. The name of the script.
sql string required. The raw SQL query for the script.
remote_host_id integer required. The remote host ID that this script will connect to.
credential_id integer required. The credential that this script will use.
parent_id integer optional. The ID of the parent job that will trigger this script.
user_context string optional. "runner" or "author", who to execute the script as when run as a template.
params array optional. An array containing the following fields:
    • name string. The variable’s name as used within your code.
    • label string. The label to present to users when asking them for the value.
    • description string. A short sentence or fragment describing this parameter to the end user.
    • type string. The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
    • required boolean. Whether this param is required.
    • value string. The value you would like to set this param to. Setting this value makes this parameter a fixed param.
    • default string. If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
- allowedValues array. The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: 'label': 'Import', 'value': 'import'

**arguments**
- list optional. Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

**schedule**
- list optional. A list containing the following elements:
  - scheduled boolean. If the item is scheduled.
  - scheduledDays array. Day based on numeric value starting at 0 for Sunday.
  - scheduledHours array. Hours of the day it is scheduled on.
  - scheduledMinutes array. Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer. Alternative to scheduled minutes, number of times to run per hour.

**notifications**
- list optional. A list containing the following elements:
  - urls array. URLs to receive a POST request at job completion
  - successEmailSubject string. Custom subject line for success e-mail.
  - successEmailBody string. Custom body text for success e-mail, written in Markdown.
  - successEmailAddresses array. Addresses to notify by e-mail when the job completes successfully.
  - successEmailFromName string. Name from which success emails are sent; defaults to "Civis."
  - successEmailReplyTo string. Address for replies to success emails; defaults to the author of the job.
  - failureEmailAddresses array. Addresses to notify by e-mail when the job fails.
  - stallWarningMinutes integer. Stall warning emails will be sent after this amount of minutes.
  - successOn boolean. If success email notifications are on.
  - failureOn boolean. If failure email notifications are on.

**next_run_at**
- string optional. The time of the next scheduled run.

**time_zone**
- string optional. The time zone of this script.

**target_project_id**
- integer optional. Target project to which script outputs will be added.

**csv_settings**
- list optional. A list containing the following elements:
  - includeHeader boolean. Whether or not to include headers in the output data. Default: true
  - compression string. The type of compression to use, if any, one of "none", "zip", or "gzip". Default: gzip
  - columnDelimiter string. Which delimiter to use, one of "comma", "tab", or "pipe". Default: comma
  - unquoted boolean. Whether or not to quote fields. Default: false
  - forceMultifile boolean. Whether or not the csv should be split into multiple files. Default: false
• `filenamePrefix` string. A user specified filename prefix for the output file to have. Default: null
• `maxFileSize` integer. The max file size, in MB, created files will be. Only available when `force_multifile` is true.

## Value
A list containing the following elements:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>id</code></td>
<td>integer. The ID for the script.</td>
</tr>
<tr>
<td><code>name</code></td>
<td>string. The name of the script.</td>
</tr>
<tr>
<td><code>type</code></td>
<td>string. The type of the script (e.g SQL, Container, Python, R, JavaScript)</td>
</tr>
<tr>
<td><code>createdAt</code></td>
<td>string. The time this script was created.</td>
</tr>
<tr>
<td><code>updatedAt</code></td>
<td>string. The time the script was last updated.</td>
</tr>
<tr>
<td><code>author</code></td>
<td>list. A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• <code>id</code> integer. The ID of this user.</td>
</tr>
<tr>
<td></td>
<td>• <code>name</code> string. This user’s name.</td>
</tr>
<tr>
<td></td>
<td>• <code>username</code> string. This user’s username.</td>
</tr>
<tr>
<td></td>
<td>• <code>initials</code> string. This user’s initials.</td>
</tr>
<tr>
<td></td>
<td>• <code>online</code> boolean. Whether this user is online.</td>
</tr>
<tr>
<td><code>state</code></td>
<td>string. The status of the script’s last run.</td>
</tr>
<tr>
<td><code>finishedAt</code></td>
<td>string. The time that the script’s last run finished.</td>
</tr>
<tr>
<td><code>category</code></td>
<td>string. The category of the script.</td>
</tr>
<tr>
<td><code>projects</code></td>
<td>array. An array containing the following fields:</td>
</tr>
<tr>
<td></td>
<td>• <code>id</code> integer. The ID for the project.</td>
</tr>
<tr>
<td></td>
<td>• <code>name</code> string. The name of the project.</td>
</tr>
<tr>
<td><code>parentId</code></td>
<td>integer. The ID of the parent job that will trigger this script</td>
</tr>
<tr>
<td><code>userContext</code></td>
<td>string. &quot;runner&quot; or &quot;author&quot;, who to execute the script as when run as a template.</td>
</tr>
<tr>
<td><code>params</code></td>
<td>array. An array containing the following fields:</td>
</tr>
<tr>
<td></td>
<td>• <code>name</code> string. The variable’s name as used within your code.</td>
</tr>
<tr>
<td></td>
<td>• <code>label</code> string. The label to present to users when asking them for the value.</td>
</tr>
<tr>
<td></td>
<td>• <code>description</code> string. A short sentence or fragment describing this parameter to the end user.</td>
</tr>
<tr>
<td></td>
<td>• <code>type</code> string. The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom</td>
</tr>
<tr>
<td></td>
<td>• <code>required</code> boolean. Whether this param is required.</td>
</tr>
<tr>
<td></td>
<td>• <code>value</code> string. The value you would like to set this param to. Setting this value makes this parameter a fixed param.</td>
</tr>
<tr>
<td></td>
<td>• <code>default</code> string. If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.</td>
</tr>
</tbody>
</table>
scripts_put_sql

- **allowedValues array**, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ‘Import’, ‘value’: ‘import’

**arguments** list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

**isTemplate** boolean, Whether others scripts use this one as a template.

**publishedAsTemplateId** integer, The ID of the template that this script is backing.

**fromTemplateId** integer, The ID of the template this script uses, if any.

**templateDependentsCount** integer, How many other scripts use this one as a template.

**templateScriptName** string, The name of the template script.

**links** list, A list containing the following elements:

- **details string**, The details link to get more information about the script.
- **runs string**, The runs link to get the run information list for this script.

**schedule** list, A list containing the following elements:

- **scheduled boolean**, If the item is scheduled.
- **scheduledDays array**, Day based on numeric value starting at 0 for Sunday.
- **scheduledHours array**, Hours of the day it is scheduled on.
- **scheduledMinutes array**, Minutes of the day it is scheduled on.
- **scheduledRunsPerHour** integer, Alternative to scheduled minutes, number of times to run per hour.

**notifications** list, A list containing the following elements:

- **urls array**, URLs to receive a POST request at job completion
- **successEmailSubject** string, Custom subject line for success e-mail.
- **successEmailBody** string, Custom body text for success e-mail, written in Markdown.
- **successEmailAddresses array**, Addresses to notify by e-mail when the job completes successfully.
- **successEmailFromName** string, Name from which success emails are sent; defaults to "Civis."
- **successEmailReplyTo** string, Address for replies to success emails; defaults to the author of the job.
- **failureEmailAddresses array**, Addresses to notify by e-mail when the job fails.
- **stallWarningMinutes** integer, Stall warning emails will be sent after this amount of minutes.
- **successOn** boolean, If success email notifications are on.
- **failureOn** boolean, If failure email notifications are on.

**runningAs** list, A list containing the following elements:

- **id integer**, The ID of this user.
- name string. This user’s name.
- username string. This user’s username.
- initials string. This user’s initials.
- online boolean. Whether this user is online.

nextRunAt string. The time of the next scheduled run.
timeZone string. The time zone of this script.
lastRun list. A list containing the following elements:
  - id integer,
  - state string,
  - createdAt string. The time that the run was queued.
  - startedAt string. The time that the run started.
  - finishedAt string. The time that the run completed.
  - error string. The error message for this run, if present.

hidden boolean. The hidden status of the item.
targetProjectId integer. Target project to which script outputs will be added.
archived string. The archival status of the requested item(s).
sql string. The raw SQL query for the script.
expandedArguments list. Expanded arguments for use in injecting into different environments.
remoteHostId integer. The remote host ID that this script will connect to.
credentialId integer. The credential that this script will use.

codePreview string. The code that this script will run with arguments inserted.
csvSettings list. A list containing the following elements:
  - includeHeader boolean. Whether or not to include headers in the output data. Default: true
  - compression string. The type of compression to use, if any, one of "none", "zip", or "gzip". Default: gzip
  - columnDelimiter string. Which delimiter to use, one of "comma", "tab", or "pipe". Default: comma
  - unquoted boolean. Whether or not to quote fields. Default: false
  - forceMultifile boolean. Whether or not the csv should be split into multiple files. Default: false
  - filenamePrefix string. A user specified filename prefix for the output file to have. Default: null
  - maxFileSize integer. The max file size, in MB, created files will be. Only available when force_multifile is true.
scripts_put_sql_archive

*Update the archive status of this object*

**Description**

Update the archive status of this object

**Usage**

```python
scripts_put_sql_archive(id, status)
```

**Arguments**

- **id** integer required. The ID of the object.
- **status** boolean required. The desired archived status of the object.

**Value**

A list containing the following elements:

- **id** integer, The ID for the script.
- **name** string, The name of the script.
- **type** string, The type of the script (e.g. SQL, Container, Python, R, JavaScript)
- **createdAt** string, The time this script was created.
- **updatedAt** string, The time the script was last updated.
- **author** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **state** string, The status of the script’s last run.
- **finishedAt** string, The time that the script’s last run finished.
- **category** string, The category of the script.
- **projects** array, An array containing the following fields:
  - **id** integer, The ID for the project.
  - **name** string, The name of the project.
- **parentId** integer, The ID of the parent job that will trigger this script
- **userContext** string, "runner" or "author", who to execute the script as when run as a template.
- **params** array, An array containing the following fields:
  - **name** string, The variable’s name as used within your code.
- label string, The label to present to users when asking them for the value.
- description string, A short sentence or fragment describing this parameter to the end user.
- type string, The type of parameter. Valid options: string, multi_line_string, integer, float, bool, file, table, database, credential_aws, credential_redshift, or credential_custom
- required boolean, Whether this param is required.
- value string, The value you would like to set this param to. Setting this value makes this parameter a fixed param.
- default string, If an argument for this parameter is not defined, it will use this default value. Use true, True, t, y, yes, or 1 for true bool’s or false, False, f, n, no, or 0 for false bool’s. Cannot be used for parameters that are required or a credential type.
- allowedValues array, The possible values this parameter can take, effectively making this an enumerable parameter. Allowed values is an array of hashes of the following format: ‘label: ’Import’, ‘value’: ’import’

arguments list, Parameter-value pairs to use when running this script. Only settable if this script has defined parameters.

isTemplate boolean, Whether others scripts use this one as a template.
publishedAsTemplateId integer, The ID of the template that this script is backing.
fromTemplateId integer, The ID of the template this script uses, if any.
templateDependentsCount integer, How many other scripts use this one as a template.
templateScriptName string, The name of the template script.

links list, A list containing the following elements:
- details string, The details link to get more information about the script.
- runs string, The runs link to get the run information list for this script.

schedule list, A list containing the following elements:
- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

notifications list, A list containing the following elements:
- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• successEmailFromName string, Name from which success emails are sent; defaults to "Civis."
• successEmailReplyTo string, Address for replies to success emails; defaults to the author of the job.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on.
• failureOn boolean, If failure email notifications are on.

runningAs list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

nextRunAt string, The time of the next scheduled run.
timeZone string, The time zone of this script.
lastRun list, A list containing the following elements:
  • id integer,
  • state string,
  • createdAt string, The time that the run was queued.
  • startedAt string, The time that the run started.
  • finishedAt string, The time that the run completed.
  • error string, The error message for this run, if present.

hidden boolean, The hidden status of the item.
targetProjectId integer, Target project to which script outputs will be added.
archived string, The archival status of the requested item(s).
sql string, The raw SQL query for the script.
expandedArguments list, Expanded arguments for use in injecting into different environments.
remoteHostId integer, The remote host ID that this script will connect to.
credentialId integer, The credential that this script will use.

• codePreview string, The code that this script will run with arguments inserted.
csvSettings list, A list containing the following elements:
  • includeHeader boolean, Whether or not to include headers in the output data. Default: true
  • compression string, The type of compression to use, if any, one of "none", "zip", or "gzip". Default: gzip
  • columnDelimiter string, Which delimiter to use, one of "comma", "tab", or "pipe". Default: comma
• unquoted boolean, Whether or not to quote fields. Default: false
• forceMultifile boolean, Whether or not the csv should be split into multiple files. Default: false
• filenamePrefix string, A user specified filename prefix for the output file to have. Default: null
• maxFileSize integer, The max file size, in MB, created files will be. Only available when force_multifile is true.

---

scripts_put_sql_git  
**Attach an item to a file in a git repo**

Description

Attach an item to a file in a git repo

Usage

```r
scripts_put_sql_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  pull_from_git = NULL
)
```

Arguments

- **id**  integer required. The ID of the file.
- **git_ref**  string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
- **git_branch**  string optional. The git branch that the file is on.
- **git_path**  string optional. The path of the file in the repository.
- **git_repo_url**  string optional. The URL of the git repository.
- **pull_from_git**  boolean optional. Automatically pull latest commit from git. Only works for scripts.

Value

A list containing the following elements:

- **gitRef**  string, A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
- **gitBranch**  string, The git branch that the file is on.
- **gitPath**  string, The path of the file in the repository.
gitRepo: list, A list containing the following elements:
  • id integer, The ID for this git repository.
  • repoUrl string, The URL for this git repository.
  • createdAt string,
  • updatedAt string,

pullFromGit: boolean, Automatically pull latest commit from git. Only works for scripts.

---

scripts_put_sql_projects

Add a SQL script to a project

Description

Add a SQL script to a project

Usage

scripts_put_sql_projects(id, project_id)

Arguments

id: integer required. The ID of the SQL script.
project_id: integer required. The ID of the project.

Value

An empty HTTP response

---

scripts_put_sql_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

scripts_put_sql_shares_groups(
  id,
  group_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
Arguments

id
integer required. The ID of the resource that is shared.

group_ids
array required. An array of one or more group IDs.

permission_level
string required. Options are: "read", "write", or "manage".

share_email_body
string optional. Custom body text for e-mail sent on a share.

send_shared_email
boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers
list, A list containing the following elements:
  • users array,
  • groups array,

writers
list, A list containing the following elements:
  • users array,
  • groups array,

owners
list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

scripts_put_sql_shares_users
Set the permissions users have on this object

Description

Set the permissions users have on this object

Usage

scripts_put_sql_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)
Arguments

id  integer required. The ID of the resource that is shared.
user_ids  array required. An array of one or more user IDs.
permission_level  string required. Options are: "read", "write", or "manage".
share_email_body  string optional. Custom body text for e-mail sent on a share.
send_shared_email  boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers  list, A list containing the following elements:
  • users array,
  • groups array,

writers  list, A list containing the following elements:
  • users array,
  • groups array,

owners  list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares  integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares  integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

search_list  Perform a search

Description

Perform a search
**Usage**

```python
search_list(
    query = NULL,
    type = NULL,
    offset = NULL,
    order = NULL,
    owner = NULL,
    limit = NULL,
    archived = NULL,
    last_run_state = NULL
)
```

**Arguments**

- `query`: string optional. The search query.
- `type`: string optional. The type for the search. It accepts a comma-separated list. Valid arguments are listed on the "GET /search/types" endpoint.
- `offset`: integer optional. The offset for the search results.
- `order`: string optional. The field on which to order the result set.
- `owner`: string optional. The owner for the search.
- `limit`: integer optional. Defaults to 10. Maximum allowed is 1000.
- `archived`: string optional. If specified, return only results with the chosen archived status; either 'true', 'false', or 'all'. Defaults to 'false'.
- `last_run_state`: string optional. The last run state of the job being searched for; either: 'queued', 'running', 'succeeded', 'failed', or 'cancelled'.

**Value**

An array containing the following fields:

- `totalResults`: integer, The number of items matching the search query.
- `aggregations`: list, Aggregations by owner and type for the search results.
- `results`: array, An array containing the following fields:
  - `score`: number, The relevance score from the search request.
  - `type`: string, The type of the item.
  - `id`: integer, The ID of the item.
  - `name`: string, The name of the item.
  - `typeName`: string, The verbose name of the type.
  - `updatedAt`: string, The time the item was last updated.
  - `owner`: string, The owner of the item.
  - `useCount`: integer, The use count of the item, if the item is a template.
  - `lastRunId`: integer, The last run id of the item, if the item is a job.
  - `lastRunState`: string, The last run state of the item, if the item is a job.
  - `lastRunStart`: string, The last run start time of the item, if the item is a job.
search_list_types

List available search types

Description

List available search types

Usage

search_list_types()

Value

An array containing the following fields:

- **type** string, The name of the item type.

services_delete

Archive a Service (deprecated, use archiving endpoints instead)

Description

Archive a Service (deprecated, use archiving endpoints instead)

Usage

services_delete(id)

Arguments

- **id** integer required.

Value

An empty HTTP response
services_delete_deployments

Delete a Service deployment

Description
Delete a Service deployment

Usage
services_delete_deployments(service_id, deployment_id)

Arguments

<table>
<thead>
<tr>
<th>service_id</th>
<th>integer required. The ID of the owning Service</th>
</tr>
</thead>
<tbody>
<tr>
<td>deployment_id</td>
<td>integer required. The ID for this deployment</td>
</tr>
</tbody>
</table>

Value
An empty HTTP response

services_delete_projects

Remove a Service from a project

Description
Remove a Service from a project

Usage
services_delete_projects(id, project_id)

Arguments

<table>
<thead>
<tr>
<th>id</th>
<th>integer required. The ID of the Service.</th>
</tr>
</thead>
<tbody>
<tr>
<td>project_id</td>
<td>integer required. The ID of the project.</td>
</tr>
</tbody>
</table>

Value
An empty HTTP response
services_delete_shares_groups

*Revoke the permissions a group has on this object*

**Description**
Revoke the permissions a group has on this object

**Usage**
services_delete_shares_groups(id, group_id)

**Arguments**
- **id**
  integer required. The ID of the resource that is shared.
- **group_id**
  integer required. The ID of the group.

**Value**
An empty HTTP response

---

services_delete_shares_users

*Revoke the permissions a user has on this object*

**Description**
Revoke the permissions a user has on this object

**Usage**
services_delete_shares_users(id, user_id)

**Arguments**
- **id**
  integer required. The ID of the resource that is shared.
- **user_id**
  integer required. The ID of the user.

**Value**
An empty HTTP response
services_delete_tokens

Revoke a token by id

Description
Revoke a token by id

Usage
services_delete_tokens(id, token_id)

Arguments
id integer required. The ID of the service.
token_id integer required. The ID of the token.

Value
An empty HTTP response

services_get
Get a Service

Description
Get a Service

Usage
services_get(id)

Arguments
id integer required.

Value
A list containing the following elements:

id integer, The ID for this Service.
name string, The name of this Service.
description string, The description of this Service.
user list, A list containing the following elements:
  • id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

type string, The type of this Service

dockerImageName string, The name of the docker image to pull from DockerHub.
dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
schedule list, A list containing the following elements:
  • runtimePlan string, Only affects the service when deployed. On Demand means that the service will be turned on when viewed and automatically turned off after periods of inactivity. Specific Times means the service will be on when scheduled. Always On means the deployed service will always be on.
  • recurrences array, List of day-hour combinations this item is scheduled for

timeZone string,
replicas integer, The number of Service replicas to deploy. When maxReplicas is set, this field defines the minimum number of replicas to deploy.
maxReplicas integer, The maximum number of Service replicas to deploy. Defining this field enables autoscaling.
instanceType string, The EC2 instance type to deploy to.
memory integer, The amount of memory allocated to each replica of the Service.
cpu integer, The amount of cpu allocated to each replica of the Service.
createdAt string,
updatedAt string,
credentials array, A list of credential IDs to pass to the Service.
apiKeyId integer, API key id of user
permissionSetId integer, The ID of the associated permission set, if any.
gitRepoUrl string, The url for the git repo where the Service code lives.
gitRepoRef string, The git reference to use when pulling code from the repo.
gitPathDir string, The path to the Service code within the git repo. If unspecified, the root directory will be used.
reportId integer, The ID of the associated report.
currentDeployment list, A list containing the following elements:
  • deploymentId integer, The ID for this deployment.
  • userId integer, The ID of the owner.
  • host string, Domain of the deployment.
  • name string, Name of the deployment.
• dockerImageName string, The name of the docker image to pull from DockerHub.
• dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
• displayUrl string, A signed URL for viewing the deployed item.
• instanceType string, The EC2 instance type requested for the deployment.
• memory integer, The memory allocated to the deployment.
• cpu integer, The cpu allocated to the deployment.
• state string, The state of the deployment.
• stateMessage string, A detailed description of the state.
• createdAt string,
• updatedAt string,
• published boolean,
• serviceId integer, The ID of owning Service

deployment list, A list containing the following elements:
• deploymentId integer, The ID for this deployment.
• userId integer, The ID of the owner.
• host string, Domain of the deployment.
• name string, Name of the deployment.
• dockerImageName string, The name of the docker image to pull from DockerHub.
• dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
• displayUrl string, A signed URL for viewing the deployed item.
• instanceType string, The EC2 instance type requested for the deployment.
• memory integer, The memory allocated to the deployment.
• cpu integer, The cpu allocated to the deployment.
• state string, The state of the deployment.
• stateMessage string, A detailed description of the state.
• createdAt string,
• updatedAt string,
• published boolean,
• serviceId integer, The ID of owning Service

currentUrl string, The URL that the service is hosted at.
previewUrl string, The URL that previews of the service are hosted at.
environmentVariables list, Environment Variables to be passed into the Service.

notifications list, A list containing the following elements:
• failureEmailAddresses array, Addresses to notify by e-mail when the service fails.
• failureOn boolean, If failure email notifications are on

archived string, The archival status of the requested item(s).
hidden boolean, The hidden status of the item.
services_get_deployments

Get details about a Service deployment

Description

Get details about a Service deployment

Usage

services_get_deployments(service_id, deployment_id)

Arguments

service_id    integer required. The ID of the owning Service
deployment_id integer required. The ID for this deployment

Value

A list containing the following elements:

- deploymentId    integer, The ID for this deployment.
- userId    integer, The ID of the owner.
- host    string, Domain of the deployment.
- name    string, Name of the deployment.
- dockerImageName    string, The name of the docker image to pull from DockerHub.
- dockerImageTag    string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl    string, A signed URL for viewing the deployed item.
- instanceType    string, The EC2 instance type requested for the deployment.
- memory    integer, The memory allocated to the deployment.
- cpu    integer, The cpu allocated to the deployment.
- state    string, The state of the deployment.
- stateMessage    string, A detailed description of the state.
- createdAt    string,
- updatedAt    string,
- published    boolean,
- serviceId    integer, The ID of owning Service
services_list  List Services

Description

List Services

Usage

services_list(
    hidden = NULL,
    archived = NULL,
    author = NULL,
    status = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)

Arguments

hidden  boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
archived  string optional. The archival status of the requested item(s).
author  string optional. If specified, return imports from this author. It accepts a comma-separated list of author IDs.
status  string optional. If specified, returns Services with one of these statuses. It accepts a comma-separated list, possible values are 'running', 'idle'.
limit  integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
page_num  integer optional. Page number of the results to return. Defaults to the first page, 1.
order  string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at.
order_dir  string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id  integer, The ID for this Service.
name  string, The name of this Service.
description  string, The description of this Service.
user list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

type string, The type of this Service
createdAt string,
updatedAt string,
gitRepoUrl string, The url for the git repo where the Service code lives.
gitRepoRef string, The git reference to use when pulling code from the repo.
gitPathDir string, The path to the Service code within the git repo. If unspecified, the root directory will be used.
currentDeployment list, A list containing the following elements:
  • deploymentId integer, The ID for this deployment.
  • userId integer, The ID of the owner.
  • host string, Domain of the deployment.
  • name string, Name of the deployment.
  • dockerImageName string, The name of the docker image to pull from DockerHub.
  • dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
  • instanceType string, The EC2 instance type requested for the deployment.
  • memory integer, The memory allocated to the deployment.
  • cpu integer, The cpu allocated to the deployment.
  • state string, The state of the deployment.
  • stateMessage string, A detailed description of the state.
  • createdAt string,
  • updatedAt string,
  • published boolean,
  • serviceId integer, The ID of owning Service
previewDeployment list, A list containing the following elements:
  • deploymentId integer, The ID for this deployment.
  • userId integer, The ID of the owner.
  • host string, Domain of the deployment.
  • name string, Name of the deployment.
  • dockerImageName string, The name of the docker image to pull from DockerHub.
  • dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
• instanceType string, The EC2 instance type requested for the deployment.
• memory integer, The memory allocated to the deployment.
• cpu integer, The cpu allocated to the deployment.
• state string, The state of the deployment.
• stateMessage string, A detailed description of the state.
• createdAt string,
• updatedAt string,
• published boolean,
• serviceId integer, The ID of owning Service

archived string, The archival status of the requested item(s).

---

services_list_deployments

*List deployments for a Service*

**Description**

List deployments for a Service

**Usage**

```python
services_list_deployments(
    service_id,
    deployment_id = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```

**Arguments**

- **service_id** integer required. The ID of the owning Service
- **deployment_id** integer optional. The ID for this deployment
- **limit** integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
- **page_num** integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order** string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at.
- **order_dir** string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.
Value

An array containing the following fields:

- deploymentId: integer, The ID for this deployment.
- userId: integer, The ID of the owner.
- host: string, Domain of the deployment.
- name: string, Name of the deployment.
- dockerImageName: string, The name of the docker image to pull from DockerHub.
- dockerImageTag: string, The tag of the docker image to pull from DockerHub (default: latest).
- instanceType: string, The EC2 instance type requested for the deployment.
- memory: integer, The memory allocated to the deployment.
- cpu: integer, The cpu allocated to the deployment.
- state: string, The state of the deployment.
- stateMessage: string, A detailed description of the state.
- createdAt: string, Created At
- updatedAt: string, Updated At
- published: boolean
- serviceId: integer, The ID of owning Service

services_list_deployments_logs

Get the logs for a Service deployment

Description

Get the logs for a Service deployment

Usage

services_list_deployments_logs(
    id, deployment_id,
    start_at = NULL,
    end_at = NULL,
    limit = NULL
)
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of the owning Service.</td>
</tr>
<tr>
<td>deployment_id</td>
<td>integer required. The ID for this deployment.</td>
</tr>
<tr>
<td>start_at</td>
<td>string optional. Log entries with a lower timestamp will be omitted.</td>
</tr>
<tr>
<td>end_at</td>
<td>string optional. Log entries with a higher timestamp will be omitted.</td>
</tr>
<tr>
<td>limit</td>
<td>integer optional. The maximum number of log messages to return. Default of 10000.</td>
</tr>
</tbody>
</table>

Value

An array containing the following fields:

- stream: string, The stream of the log. One of "stdout", "stderr".
- createdAt: string, The time the log was created.
- source: string, The source of the log. One of "system", "user".

services_list_projects

List the projects a Service belongs to

Description

List the projects a Service belongs to

Usage

services_list_projects(id, hidden = NULL)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of the Service.</td>
</tr>
<tr>
<td>hidden</td>
<td>boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.</td>
</tr>
</tbody>
</table>

Value

An array containing the following fields:

- id: integer, The ID for this project.
- author: list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
List users and groups permissioned on this object

**Description**

List users and groups permissioned on this object

**Usage**

```
services_list_shares(id)
```

**Arguments**

- **id**
  
  integer required. The ID of the resource that is shared.

**Value**

An array containing the following fields:

- **readers**
  
  list, A list containing the following elements:
  
  - users array,
  
  - groups array,

- **writers**
  
  list, A list containing the following elements:
  
  - users array,
  
  - groups array,

- **owners**
  
  list, A list containing the following elements:
  
  - users array,
- groups array,

**totalUserShares**
integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

**totalGroupShares**
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**services_list_tokens**  
*List tokens*

---

**Description**

List tokens

**Usage**

`services_list_tokens(id)`

**Arguments**

- **id**  
  integer required. The ID of the service.

**Value**

An array containing the following fields:

- **id**  
  integer, The ID of the token.

- **name**  
  string, The name of the token.

- **user**  
  list, A list containing the following elements:
    - **id**  
      integer, The ID of this user.
    - **name**  
      string, This user’s name.
    - **username**  
      string, This user’s username.
    - **initials**  
      string, This user’s initials.
    - **online**  
      boolean, Whether this user is online.

- **machineToken**  
  boolean, If true, this token is not tied to a particular user.

- **createdAt**  
  string, The date and time when the token was created.
services_patch | *Update some attributes of this Service*

**Description**

Update some attributes of this Service

**Usage**

```python
def services_patch(
    id,
    name = NULL,
    description = NULL,
    docker_image_name = NULL,
    docker_image_tag = NULL,
    schedule = NULL,
    replicas = NULL,
    max_replicas = NULL,
    instance_type = NULL,
    memory = NULL,
    cpu = NULL,
    credentials = NULL,
    api_key_id = NULL,
    permission_set_id = NULL,
    git_repo_url = NULL,
    git_repo_ref = NULL,
    git_path_dir = NULL,
    environment_variables = NULL,
    notifications = NULL
)
```

**Arguments**

- `id` integer required. The ID for this Service.
- `name` string optional. The name of this Service.
- `description` string optional. The description of this Service.
- `docker_image_name` string optional. The name of the docker image to pull from DockerHub.
- `docker_image_tag` string optional. The tag of the docker image to pull from DockerHub (default: latest).
- `schedule` list optional. A list containing the following elements:
  - `runtimePlan` string. Only affects the service when deployed. On Demand means that the service will be turned on when viewed and automatically turned off after periods of inactivity. Specific Times means the service will
be on when scheduled. Always On means the deployed service will always be on.

• recurrences array. List of day-hour combinations this item is scheduled for

replicas integer optional. The number of Service replicas to deploy. When maxReplicas is set, this field defines the minimum number of replicas to deploy.

max_replicas integer optional. The maximum number of Service replicas to deploy. Defining this field enables autoscaling.

instance_type string optional. The EC2 instance type to deploy to.

memory integer optional. The amount of memory allocated to each replica of the Service.

cpu integer optional. The amount of cpu allocated to each replica of the Service.

credentials array optional. A list of credential IDs to pass to the Service.

api_key_id integer optional. API key id of user

permission_set_id integer optional. The ID of the associated permission set, if any.

git_repo_url string optional. The url for the git repo where the Service code lives.

git_repo_ref string optional. The git reference to use when pulling code from the repo.

git_path_dir string optional. The path to the Service code within the git repo. If unspecified, the root directory will be used.

environment_variables list optional. Environment Variables to be passed into the Service.

notifications list optional. A list containing the following elements:

• failureEmailAddresses array, Addresses to notify by e-mail when the service fails.

• failureOn boolean, If failure email notifications are on

Value

A list containing the following elements:

id integer, The ID for this Service.

name string, The name of this Service.

description string, The description of this Service.

user list, A list containing the following elements:

• id integer, The ID of this user.

• name string, This user’s name.

• username string, This user’s username.

• initials string, This user’s initials.

• online boolean, Whether this user is online.

type string, The type of this Service

dockerImageName string, The name of the docker image to pull from DockerHub.

dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
services_patch

schedule list, A list containing the following elements:
  - runtimePlan string, Only affects the service when deployed. On Demand means that the service will be turned on when viewed and automatically turned off after periods of inactivity. Specific Times means the service will be on when scheduled. Always On means the deployed service will always be on.
  - recurrences array, List of day-hour combinations this item is scheduled for

timeZone string,
replicas integer, The number of Service replicas to deploy. When maxReplicas is set, this field defines the minimum number of replicas to deploy.
maxReplicas integer, The maximum number of Service replicas to deploy. Defining this field enables autoscaling.
instanceType string, The EC2 instance type to deploy to.
memory integer, The amount of memory allocated to each replica of the Service.
cpu integer, The amount of cpu allocated to each replica of the Service.
createdAt string,
updatedAt string,
credentials array, A list of credential IDs to pass to the Service.
apiKeyId integer, API key id of user
permissionSetId integer, The ID of the associated permission set, if any.
gitRepoUrl string, The url for the git repo where the Service code lives.
gitRepoRef string, The git reference to use when pulling code from the repo.
gitPathDir string, The path to the Service code within the git repo. If unspecified, the root directory will be used.
reportId integer, The ID of the associated report.
currentDeployment list, A list containing the following elements:
  - deploymentId integer, The ID for this deployment.
  - userId integer, The ID of the owner.
  - host string, Domain of the deployment.
  - name string, Name of the deployment.
  - dockerImageName string, The name of the docker image to pull from DockerHub.
  - dockerImageTag string, The tag of the docker image to pull from Docker-Hub (default: latest).
  - displayUrl string, A signed URL for viewing the deployed item.
  - instanceType string, The EC2 instance type requested for the deployment.
  - memory integer, The memory allocated to the deployment.
  - cpu integer, The cpu allocated to the deployment.
  - state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- createdAt string,
- updatedAt string,
- published boolean,
- serviceId integer, The ID of owning Service

**previewDeployment**

list, A list containing the following elements:

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment.
- cpu integer, The cpu allocated to the deployment.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- createdAt string,
- updatedAt string,
- published boolean,
- serviceId integer, The ID of owning Service

**currentUrl** string, The URL that the service is hosted at.

**previewUrl** string, The URL that previews of the service are hosted at.

**environmentVariables**

list, Environment Variables to be passed into the Service.

**notifications**

list, A list containing the following elements:

- failureEmailAddresses array, Addresses to notify by e-mail when the service fails.
- failureOn boolean, If failure email notifications are on

**archived** string, The archival status of the requested item(s).

**hidden** boolean, The hidden status of the item.
Create a Service

Usage

```python
services_post(
    name = NULL,
    description = NULL,
    type = NULL,
    docker_image_name = NULL,
    docker_image_tag = NULL,
    schedule = NULL,
    replicas = NULL,
    max_replicas = NULL,
    instance_type = NULL,
    memory = NULL,
    cpu = NULL,
    credentials = NULL,
    api_key_id = NULL,
    permission_set_id = NULL,
    git_repo_url = NULL,
    git_repo_ref = NULL,
    git_path_dir = NULL,
    environment_variables = NULL,
    notifications = NULL,
    hidden = NULL
)
```

Arguments

- **name** (string optional): The name of this Service.
- **description** (string optional): The description of this Service.
- **type** (string optional): The type of this Service.
- **docker_image_name** (string optional): The name of the docker image to pull from DockerHub.
- **docker_image_tag** (string optional): The tag of the docker image to pull from DockerHub (default: latest).
- **schedule** (list optional): A list containing the following elements:
  - **runtimePlan** (string): Only affects the service when deployed. On Demand means that the service will be turned on when viewed and automatically
turned off after periods of inactivity. Specific Times means the service will be on when scheduled. Always On means the deployed service will always be on.

- recurrences array, List of day-hour combinations this item is scheduled for

replicas integer optional. The number of Service replicas to deploy. When maxReplicas is set, this field defines the minimum number of replicas to deploy.

max_replicas integer optional. The maximum number of Service replicas to deploy. Defining this field enables autoscaling.

instance_type string optional. The EC2 instance type to deploy to.

memory integer optional. The amount of memory allocated to each replica of the Service.

cpu integer optional. The amount of cpu allocated to each replica of the the Service.

credentials array optional. A list of credential IDs to pass to the Service.

api_key_id integer optional. API key id of user

permission_set_id integer optional. The ID of the associated permission set, if any.

git_repo_url string optional. The url for the git repo where the Service code lives.

git_repo_ref string optional. The git reference to use when pulling code from the repo.

git_path_dir string optional. The path to the Service code within the git repo. If unspecified, the root directory will be used.

environment_variables list optional. Environment Variables to be passed into the Service.

notifications list optional. A list containing the following elements:
  - failureEmailAddresses array, Addresses to notify by e-mail when the service fails.
  - failureOn boolean, If failure email notifications are on

hidden boolean optional. The hidden status of the item.

**Value**

A list containing the following elements:

id integer, The ID for this Service.

name string, The name of this Service.

description string, The description of this Service.

user list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.

type string, The type of this Service
**dockerImageName**
string, The name of the docker image to pull from DockerHub.

**dockerImageTag**
string, The tag of the docker image to pull from DockerHub (default: latest).

**schedule**
list, A list containing the following elements:
- **runtimePlan**
  string, Only affects the service when deployed. On Demand means that the service will be turned on when viewed and automatically turned off after periods of inactivity. Specific Times means the service will be on when scheduled. Always On means the deployed service will always be on.
- **recurrences**
  array, List of day-hour combinations this item is scheduled for

**timeZone**
string,

**replicas**
integer, The number of Service replicas to deploy. When maxReplicas is set, this field defines the minimum number of replicas to deploy.

**maxReplicas**
integer, The maximum number of Service replicas to deploy. Defining this field enables autoscaling.

**instanceType**
string, The EC2 instance type to deploy to.

**memory**
integer, The amount of memory allocated to each replica of the Service.

**cpu**
integer, The amount of cpu allocated to each replica of the Service.

**createdAt**
string,

**updatedAt**
string,

**credentials**
array, A list of credential IDs to pass to the Service.

**apiKeyId**
integer, API key id of user

**permissionSetId**
integer, The ID of the associated permission set, if any.

**gitRepoUrl**
string, The url for the git repo where the Service code lives.

**gitRepoRef**
string, The git reference to use when pulling code from the repo.

**gitPathDir**
string, The path to the Service code within the git repo. If unspecified, the root directory will be used.

**reportId**
integer, The ID of the associated report.

**currentDeployment**
list, A list containing the following elements:
- **deploymentId**
  integer, The ID for this deployment.
- **userId**
  integer, The ID of the owner.
- **host**
  string, Domain of the deployment.
- **name**
  string, Name of the deployment.
- **dockerImageName**
  string, The name of the docker image to pull from DockerHub.
- **dockerImageTag**
  string, The tag of the docker image to pull from DockerHub (default: latest).
- **displayUrl**
  string, A signed URL for viewing the deployed item.
- **instanceType**
  string, The EC2 instance type requested for the deployment.
• memory integer, The memory allocated to the deployment.
• cpu integer, The cpu allocated to the deployment.
• state string, The state of the deployment.
• stateMessage string, A detailed description of the state.
• createdAt string,
• updatedAt string,
• published boolean,
• serviceId integer, The ID of owning Service

previewDeployment
• deploymentId integer, The ID for this deployment.
• userId integer, The ID of the owner.
• host string, Domain of the deployment.
• name string, Name of the deployment.
• dockerImageName string, The name of the docker image to pull from DockerHub.
• dockerImageTag string, The tag of the docker image to pull from Docker-Hub (default: latest).
• displayUrl string, A signed URL for viewing the deployed item.
• instanceType string, The EC2 instance type requested for the deployment.
• memory integer, The memory allocated to the deployment.
• cpu integer, The cpu allocated to the deployment.
• state string, The state of the deployment.
• stateMessage string, A detailed description of the state.
• createdAt string,
• updatedAt string,
• published boolean,
• serviceId integer, The ID of owning Service

currentUrl string, The URL that the service is hosted at.
previewUrl string, The URL that previews of the service are hosted at.
environmentVariables
• list, Environment Variables to be passed into the Service.

notifications
• list, A list containing the following elements:
  • failureEmailAddresses array, Addresses to notify by e-mail when the service fails.
  • failureOn boolean, If failure email notifications are on

archived string, The archival status of the requested item(s).
hidden boolean, The hidden status of the item.
**services_post_clone**  

*Clone this Service*

**Description**

Clone this Service

**Usage**

```
services_post_clone(id)
```

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required.</td>
</tr>
</tbody>
</table>

**Value**

A list containing the following elements:

- **id** integer, The ID for this Service.
- **name** string, The name of this Service.
- **description** string, The description of this Service.
- **user** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **type** string, The type of this Service
- **dockerImageName** string, The name of the docker image to pull from DockerHub.
- **dockerImageTag** string, The tag of the docker image to pull from DockerHub (default: latest).
- **schedule** list, A list containing the following elements:
  - **runtimePlan** string, Only affects the service when deployed. On Demand means that the service will be turned on when viewed and automatically turned off after periods of inactivity. Specific Times means the service will be on when scheduled. Always On means the deployed service will always be on.
  - **recurrences** array, List of day-hour combinations this item is scheduled for
- **timeZone** string,
- **replicas** integer, The number of Service replicas to deploy. When maxReplicas is set, this field defines the minimum number of replicas to deploy.
maxReplicas integer, The maximum number of Service replicas to deploy. Defining this field enables autoscaling.

instanceType string, The EC2 instance type to deploy to.

memory integer, The amount of memory allocated to each replica of the Service.

cpu integer, The amount of cpu allocated to each replica of the Service.

createdAt string,

updatedAt string,

credentials array, A list of credential IDs to pass to the Service.

apiKeyId integer, API key id of user

permissionSetId integer, The ID of the associated permission set, if any.

gitRepoUrl string, The url for the git repo where the Service code lives.

gitRepoRef string, The git reference to use when pulling code from the repo.

gitPathDir string, The path to the Service code within the git repo. If unspecified, the root directory will be used.

reportId integer, The ID of the associated report.

currentDeployment list, A list containing the following elements:
  • deploymentId integer, The ID for this deployment.
  • userId integer, The ID of the owner.
  • host string, Domain of the deployment.
  • name string, Name of the deployment.
  • dockerImageName string, The name of the docker image to pull from DockerHub.
  • dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
  • displayUrl string, A signed URL for viewing the deployed item.
  • instanceType string, The EC2 instance type requested for the deployment.
  • memory integer, The memory allocated to the deployment.
  • cpu integer, The cpu allocated to the deployment.
  • state string, The state of the deployment.
  • stateMessage string, A detailed description of the state.
  • createdAt string,
  • updatedAt string,
  • published boolean,
  • serviceId integer, The ID of owning Service

previewDeployment list, A list containing the following elements:
  • deploymentId integer, The ID for this deployment.
  • userId integer, The ID of the owner.
  • host string, Domain of the deployment.
• name string, Name of the deployment.
• dockerImageName string, The name of the docker image to pull from DockerHub.
• dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
• displayUrl string, A signed URL for viewing the deployed item.
• instanceType string, The EC2 instance type requested for the deployment.
• memory integer, The memory allocated to the deployment.
• cpu integer, The cpu allocated to the deployment.
• state string, The state of the deployment.
• stateMessage string, A detailed description of the state.
• createdAt string,
• updatedAt string,
• published boolean,
• serviceId integer, The ID of owning Service

currentUrl string, The URL that the service is hosted at.
previewUrl string, The URL that previews of the service are hosted at.
environmentVariables list, Environment Variables to be passed into the Service.
notifications list, A list containing the following elements:
  • failureEmailAddresses array, Addresses to notify by e-mail when the service fails.
  • failureOn boolean, If failure email notifications are on
archived string, The archival status of the requested item(s).
hidden boolean, The hidden status of the item.

---

services_post_deployments

Deploy a Service

Description

Deploy a Service

Usage

services_post_deployments(service_id, deployment_id = NULL, published = NULL)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>service_id</td>
<td>integer required. The ID of the owning Service</td>
</tr>
<tr>
<td>deployment_id</td>
<td>integer optional. The ID for this deployment</td>
</tr>
<tr>
<td>published</td>
<td>boolean optional.</td>
</tr>
</tbody>
</table>
services_post_redeploy

Redeploy a Service

Description
Redeploy a Service

Usage
services_post_redeploy(service_id, deployment_id = NULL, published = NULL)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>service_id</td>
<td>integer required. The ID of the owning Service</td>
</tr>
<tr>
<td>deployment_id</td>
<td>integer optional. The ID for this deployment</td>
</tr>
<tr>
<td>published</td>
<td>boolean optional.</td>
</tr>
</tbody>
</table>
Value

A list containing the following elements:

- deploymentId: integer, The ID for this deployment.
- userId: integer, The ID of the owner.
- host: string, Domain of the deployment.
- name: string, Name of the deployment.
- dockerImageName: string, The name of the docker image to pull from DockerHub.
- dockerImageTag: string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl: string, A signed URL for viewing the deployed item.
- instanceType: string, The EC2 instance type requested for the deployment.
- memory: integer, The memory allocated to the deployment.
- cpu: integer, The cpu allocated to the deployment.
- state: string, The state of the deployment.
- stateMessage: string, A detailed description of the state.
- createdAt: string,
- updatedAt: string,
- published: boolean,
- serviceId: integer, The ID of owning Service

services_post_tokens  Create a new long-lived service token

Description

Create a new long-lived service token

Usage

services_post_tokens(id, name, machine_token = NULL)

Arguments

id: integer required. The ID of the service.
name: string required. The name of the token.
machine_token: boolean optional. If true, create a compact token with no user information.
Value

A list containing the following elements:

- **id**: integer, The ID of the token.
- **name**: string, The name of the token.
- **user**: list, A list containing the following elements:
  - **id**: integer, The ID of this user.
  - **name**: string, This user’s name.
  - **username**: string, This user’s username.
  - **initials**: string, This user’s initials.
  - **online**: boolean, Whether this user is online.
- **machineToken**: boolean, If true, this token is not tied to a particular user.
- **createdAt**: string, The date and time when the token was created.
- **token**: string, The value of the token. Only returned when the token is first created.

```python
services_put

Replace all attributes of this Service
```

Description

Replace all attributes of this Service

Usage

```python
services_put(
    id,
    name = NULL,
    description = NULL,
    docker_image_name = NULL,
    docker_image_tag = NULL,
    schedule = NULL,
    replicas = NULL,
    max_replicas = NULL,
    instance_type = NULL,
    memory = NULL,
    cpu = NULL,
    credentials = NULL,
    api_key_id = NULL,
    permission_set_id = NULL,
    git_repo_url = NULL,
    git_repo_ref = NULL,
    git_path_dir = NULL,
    environment_variables = NULL,
    notifications = NULL
)
```
Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID for this Service.</td>
</tr>
<tr>
<td>name</td>
<td>string optional. The name of this Service.</td>
</tr>
<tr>
<td>description</td>
<td>string optional. The description of this Service.</td>
</tr>
<tr>
<td>docker_image_name</td>
<td>string optional. The name of the docker image to pull from DockerHub.</td>
</tr>
<tr>
<td>docker_image_tag</td>
<td>string optional. The tag of the docker image to pull from DockerHub (default: latest).</td>
</tr>
<tr>
<td>schedule</td>
<td>list optional. A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• runtimePlan string. Only affects the service when deployed. On Demand means that the service will be turned on when viewed and automatically turned off after periods of inactivity. Specific Times means the service will be on when scheduled. Always On means the deployed service will always be on.</td>
</tr>
<tr>
<td></td>
<td>• recurrences array. List of day-hour combinations this item is scheduled for</td>
</tr>
<tr>
<td>replicas</td>
<td>integer optional. The number of Service replicas to deploy. When maxReplicas is set, this field defines the minimum number of replicas to deploy.</td>
</tr>
<tr>
<td>max_replicas</td>
<td>integer optional. The maximum number of Service replicas to deploy. Defining this field enables autoscaling.</td>
</tr>
<tr>
<td>instance_type</td>
<td>string optional. The EC2 instance type to deploy to.</td>
</tr>
<tr>
<td>memory</td>
<td>integer optional. The amount of memory allocated to each replica of the Service.</td>
</tr>
<tr>
<td>cpu</td>
<td>integer optional. The amount of cpu allocated to each replica of the Service.</td>
</tr>
<tr>
<td>credentials</td>
<td>array optional. A list of credential IDs to pass to the Service.</td>
</tr>
<tr>
<td>api_key_id</td>
<td>integer optional. API key id of user</td>
</tr>
<tr>
<td>permission_set_id</td>
<td>integer optional. The ID of the associated permission set, if any.</td>
</tr>
<tr>
<td>git_repo_url</td>
<td>string optional. The url for the git repo where the Service code lives.</td>
</tr>
<tr>
<td>git_repo_ref</td>
<td>string optional. The git reference to use when pulling code from the repo.</td>
</tr>
<tr>
<td>git_path_dir</td>
<td>string optional. The path to the Service code within the git repo. If unspecified, the root directory will be used.</td>
</tr>
<tr>
<td>environment_variables</td>
<td>list optional. Environment Variables to be passed into the Service.</td>
</tr>
<tr>
<td>notifications</td>
<td>list optional. A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• failureEmailAddresses array, Addresses to notify by e-mail when the service fails.</td>
</tr>
<tr>
<td></td>
<td>• failureOn boolean, If failure email notifications are on</td>
</tr>
</tbody>
</table>
Value

A list containing the following elements:

- **id** integer, The ID for this Service.
- **name** string, The name of this Service.
- **description** string, The description of this Service.
- **user** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **type** string, The type of this Service.
- **dockerImageName** string, The name of the docker image to pull from DockerHub.
- **dockerImageTag** string, The tag of the docker image to pull from DockerHub (default: latest).
- **schedule** list, A list containing the following elements:
  - **runtimePlan** string, Only affects the service when deployed. On Demand means that the service will be turned on when viewed and automatically turned off after periods of inactivity. Specific Times means the service will be on when scheduled. Always On means the deployed service will always be on.
  - **recurrences** array, List of day-hour combinations this item is scheduled for.
- **timeZone** string,
- **replicas** integer, The number of Service replicas to deploy. When maxReplicas is set, this field defines the minimum number of replicas to deploy.
- **maxReplicas** integer, The maximum number of Service replicas to deploy. Defining this field enables autoscaling.
- **instanceType** string, The EC2 instance type to deploy to.
- **memory** integer, The amount of memory allocated to each replica of the Service.
- **cpu** integer, The amount of cpu allocated to each replica of the Service.
- **createdAt** string,
- **updatedAt** string,
- **credentials** array, A list of credential IDs to pass to the Service.
- **apiKeyId** integer, API key id of user
- **permissionSetId** integer, The ID of the associated permission set, if any.
- **gitRepoUrl** string, The url for the git repo where the Service code lives.
- **gitRepoRef** string, The git reference to use when pulling code from the repo.
- **gitPathDir** string, The path to the Service code within the git repo. If unspecified, the root directory will be used.
services_put

**reportId**
integer, The ID of the associated report.

**currentDeployment**
list, A list containing the following elements:
- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment.
- cpu integer, The cpu allocated to the deployment.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- createdAt string,
- updatedAt string,
- published boolean,
- serviceId integer, The ID of owning Service

**previewDeployment**
list, A list containing the following elements:
- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment.
- cpu integer, The cpu allocated to the deployment.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- createdAt string,
- updatedAt string,
- published boolean,
- serviceId integer, The ID of owning Service

**currentUrl**
string, The URL that the service is hosted at.
services_put_archive  

**Description**

Update the archive status of this object

**Usage**

services_put_archive(id, status)

**Arguments**

- **id** integer required. The ID of the object.
- **status** boolean required. The desired archived status of the object.

**Value**

A list containing the following elements:

- **id** integer, The ID for this Service.
- **name** string, The name of this Service.
- **description** string, The description of this Service.
- **user** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **type** string, The type of this Service
- **dockerImageName** string, The name of the docker image to pull from DockerHub.
- **dockerImageTag** string, The tag of the docker image to pull from DockerHub (default: latest).
schedule list, A list containing the following elements:
  • runtimePlan string, Only affects the service when deployed. On Demand means that the service will be turned on when viewed and automatically turned off after periods of inactivity. Specific Times means the service will be on when scheduled. Always On means the deployed service will always be on.
  • recurrences array, List of day-hour combinations this item is scheduled for

timeZone string,
replicas integer, The number of Service replicas to deploy. When maxReplicas is set, this field defines the minimum number of replicas to deploy.
maxReplicas integer, The maximum number of Service replicas to deploy. Defining this field enables autoscaling.
instanceType string, The EC2 instance type to deploy to.
memory integer, The amount of memory allocated to each replica of the Service.
cpu integer, The amount of cpu allocated to each replica of the Service.
createdAt string,
updatedAt string,
credentials array, A list of credential IDs to pass to the Service.
apiKeyId integer, API key id of user
permissionSetId integer, The ID of the associated permission set, if any.
gitRepoUrl string, The url for the git repo where the Service code lives.
gitRepoRef string, The git reference to use when pulling code from the repo.
gitPathDir string, The path to the Service code within the git repo. If unspecified, the root directory will be used.
reportId integer, The ID of the associated report.
currentDeployment list, A list containing the following elements:
  • deploymentId integer, The ID for this deployment.
  • userId integer, The ID of the owner.
  • host string, Domain of the deployment.
  • name string, Name of the deployment.
  • dockerImageName string, The name of the docker image to pull from DockerHub.
  • dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
  • displayUrl string, A signed URL for viewing the deployed item.
  • instanceType string, The EC2 instance type requested for the deployment.
  • memory integer, The memory allocated to the deployment.
  • cpu integer, The cpu allocated to the deployment.
  • state string, The state of the deployment.
services_put_archive

- stateMessage string, A detailed description of the state.
- createdAt string,
- updatedAt string,
- published boolean,
- serviceId integer, The ID of owning Service

previewDeployment

- deploymentId integer, The ID for this deployment.
- userId integer, The ID of the owner.
- host string, Domain of the deployment.
- name string, Name of the deployment.
- dockerImageName string, The name of the docker image to pull from DockerHub.
- dockerImageTag string, The tag of the docker image to pull from DockerHub (default: latest).
- displayUrl string, A signed URL for viewing the deployed item.
- instanceType string, The EC2 instance type requested for the deployment.
- memory integer, The memory allocated to the deployment.
- cpu integer, The cpu allocated to the deployment.
- state string, The state of the deployment.
- stateMessage string, A detailed description of the state.
- createdAt string,
- updatedAt string,
- published boolean,
- serviceId integer, The ID of owning Service

currentUrl string, The URL that the service is hosted at.

previewUrl string, The URL that previews of the service are hosted at.

environmentVariables list, Environment Variables to be passed into the Service.

notifications list, A list containing the following elements:

- failureEmailAddresses array, Addresses to notify by e-mail when the service fails.
- failureOn boolean, If failure email notifications are on

archived string, The archival status of the requested item(s).

hidden boolean, The hidden status of the item.
services_put_projects  Add a Service to a project

Description

Add a Service to a project

Usage

services_put_projects(id, project_id)

Arguments

id  integer required. The ID of the Service.
project_id  integer required. The ID of the project.

Value

An empty HTTP response

services_put_shares_groups  Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

services_put_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
Arguments

**id**
integer required. The ID of the resource that is shared.

**group_ids**
array required. An array of one or more group IDs.

**permission_level**
string required. Options are: "read", "write", or "manage".

**share_email_body**
string optional. Custom body text for e-mail sent on a share.

**send_shared_email**
boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- **readers**
  list, A list containing the following elements:
  - **users** array,
  - **groups** array,

- **writers**
  list, A list containing the following elements:
  - **users** array,
  - **groups** array,

- **owners**
  list, A list containing the following elements:
  - **users** array,
  - **groups** array,

- **totalUserShares**
  integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

- **totalGroupShares**
  integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**services_put_shares_users**

*Set the permissions users have on this object*

---

Description

Set the permissions users have on this object

Usage

```python
services_put_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```
Arguments

- **id** integer required. The ID of the resource that is shared.
- **user_ids** array required. An array of one or more user IDs.
- **permission_level** string required. Options are: "read", "write", or "manage".
- **share_email_body** string optional. Custom body text for e-mail sent on a share.
- **send_shared_email** boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,
- **writers** list, A list containing the following elements:
  - users array,
  - groups array,
- **owners** list, A list containing the following elements:
  - users array,
  - groups array,
- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**Description**

SQL escaping

**Usage**

```r
sql(...)```

**Arguments**

```
... Character vectors that will be combined into a single SQL expression.
```
stash_local_dataframe  
Stash a data frame in feather or csv format, depending on CivisML version.

Description
Stash a data frame in feather or csv format, depending on CivisML version.

Usage
stash_local_dataframe(x, tmpl_id)

Arguments
- x: data.frame to stash
- tmpl_id: CivisML training template id

Value
file id where dataframe is stored

storage_hosts_delete  
Delete a storage host (deprecated)

Description
Delete a storage host (deprecated)

Usage
storage_hosts_delete(id)

Arguments
- id: integer required. The ID of the storage host.

Value
An empty HTTP response
storage_hosts_delete_shares_groups

Revoke the permissions a group has on this object

Description
Revoke the permissions a group has on this object

Usage
storage_hosts_delete_shares_groups(id, group_id)

Arguments
id integer required. The ID of the resource that is shared.
group_id integer required. The ID of the group.

Value
An empty HTTP response

storage_hosts_delete_shares_users

Revoke the permissions a user has on this object

Description
Revoke the permissions a user has on this object

Usage
storage_hosts_delete_shares_users(id, user_id)

Arguments
id integer required. The ID of the resource that is shared.
user_id integer required. The ID of the user.

Value
An empty HTTP response
storage_hosts_get  
*Get a storage host*

**Description**

Get a storage host

**Usage**

```python
storage_hosts_get(id)
```

**Arguments**

- `id`  
  integer required. The ID of the storage host.

**Value**

A list containing the following elements:

- `id`  
  integer, The ID of the storage host.

- `owner`  
  list, A list containing the following elements:
  - `id`  
    integer, The ID of this user.
  - `name` string, This user’s name.
  - `username` string, This user’s username.
  - `initials` string, This user’s initials.
  - `online` boolean, Whether this user is online.

- `name` string, The human readable name for the storage host.

- `provider` string, The storage provider. One of: s3.

- `bucket` string, The bucket for this storage host.

- `s3Options` list, A list containing the following elements:
  - `region` string, The region for this storage host (ex. "us-east-1")

---

storage_hosts_list  
*List the storage hosts*

**Description**

List the storage hosts

**Usage**

```python
storage_hosts_list()
```
**Value**

An array containing the following fields:

- **id** integer, The ID of the storage host.
- **owner** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **name** string, The human readable name for the storage host.
- **provider** string, The storage provider. One of: s3.
- **bucket** string, The bucket for this storage host.
- **s3Options** list, A list containing the following elements:
  - **region** string, The region for this storage host (ex. "us-east-1")

---

**storage_hosts_list_shares**

*List users and groups permissioned on this object*

---

**Description**

List users and groups permissioned on this object.

**Usage**

`storage_hosts_list_shares(id)`

**Arguments**

- **id** integer required. The ID of the resource that is shared.

**Value**

An array containing the following fields:

- **readers** list, A list containing the following elements:
  - **users** array,
  - **groups** array,
- **writers** list, A list containing the following elements:
  - **users** array,
  - **groups** array,
- **owners** list, A list containing the following elements:
storage_hosts_patch

• users array,
• groups array,

totalUserShares
  integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares
  integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

storage_hosts_patch  Update some attributes of this storage host

Description
Update some attributes of this storage host

Usage
storage_hosts_patch(
  id,
  name = NULL,
  provider = NULL,
  bucket = NULL,
  s3_options = NULL
)

Arguments
id  integer required. The ID of the storage host.
name  string optional. The human readable name for the storage host.
provider  string optional. The storage provider. One of: s3.
bucket  string optional. The bucket for this storage host.
s3_options  list optional. A list containing the following elements:
  • region string. The region for this storage host (ex. "us-east-1")

Value
A list containing the following elements:
id  integer, The ID of the storage host.
owner  list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string. This user's name.
  • username string. This user's username.
  • initials string, This user's initials.
Description

Create a new storage host

Usage

```
storage_hosts_post(provider, bucket, name, s3_options = NULL)
```

Arguments

- **provider** string required. The storage provider. One of: s3.
- **bucket** string required. The bucket for this storage host.
- **name** string required. The human readable name for the storage host.
- **s3_options** list optional. A list containing the following elements:
  - **region** string, The region for this storage host (ex. "us-east-1")

Value

A list containing the following elements:

- **id** integer, The ID of the storage host.
- **owner** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **name** string, The human readable name for the storage host.
- **provider** string, The storage provider. One of: s3.
- **bucket** string, The bucket for this storage host.
- **s3_options** list, A list containing the following elements:
  - **region** string, The region for this storage host (ex. "us-east-1")
storage_hosts_put  Replace all attributes of this storage host

Description
Replace all attributes of this storage host

Usage
storage_hosts_put(id, name, provider, bucket, s3_options = NULL)

Arguments
id  integer required. The ID of the storage host.
name  string required. The human readable name for the storage host.
provider  string required. The storage provider. One of: s3.
bucket  string required. The bucket for this storage host.
s3_options  list optional. A list containing the following elements:
  • region string, The region for this storage host (ex. "us-east-1")

Value
A list containing the following elements:
id  integer, The ID of the storage host.
owner  list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
name  string, The human readable name for the storage host.
provider  string, The storage provider. One of: s3.
bucket  string, The bucket for this storage host.
s3Options  list, A list containing the following elements:
  • region string, The region for this storage host (ex. "us-east-1")
storage_hosts_put_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

storage_hosts_put_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)

Arguments

id integer required. The ID of the resource that is shared.
group_ids array required. An array of one or more group IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers list, A list containing the following elements:
    • users array,
    • groups array,

writers list, A list containing the following elements:
    • users array,
    • groups array,

owners list, A list containing the following elements:
    • users array,
    • groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares
integer. For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

storage_hosts_put_shares_users
Set the permissions users have on this object

Description
Set the permissions users have on this object

Usage
storage_hosts_put_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)

Arguments
id integer required. The ID of the resource that is shared.
user_ids array required. An array of one or more user IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value
A list containing the following elements:

readers list, A list containing the following elements:
  • users array,
  • groups array,

writers list, A list containing the following elements:
  • users array,
  • groups array,

owners list, A list containing the following elements:
  • users array,
tables_delete_projects

Remove a Table from a project

Description

Remove a Table from a project

Usage

tables_delete_projects(id, project_id)

Arguments

id        integer required. The ID of the Table.
project_id integer required. The ID of the project.

Value

An empty HTTP response

tables_get Show basic table info

Description

Show basic table info

Usage

tables_get(id)

Arguments

id        integer required.
Value

A list containing the following elements:

- id
  integer, The ID of the table.
- databaseId
  integer, The ID of the database.
- schema
  string, The name of the schema containing the table.
- name
  string, Name of the table.
- description
  string, The description of the table, as specified by the table owner
- isView
  boolean, True if this table represents a view. False if it represents a regular table.
- rowCount
  integer, The number of rows in the table.
- columnCount
  integer, The number of columns in the table.
- sizeMb
  number, The size of the table in megabytes.
- owner
  string, The database username of the table’s owner.
- distkey
  string, The column used as the Amazon Redshift distkey.
- sortkeys
  string, The column used as the Amazon Redshift sortkey.
- refreshStatus
  string, How up-to-date the table’s statistics on row counts, null counts, distinct counts, and values distributions are. One of: refreshing, stale, or current.
- lastRefresh
  string, The time of the last statistics refresh.
- dataUpdatedAt
  string, The last time that Civis Platform captured a change in this table. Only applicable for Redshift tables; please see the Civis help desk for more info.
- schemaUpdatedAt
  string, The last time that Civis Platform captured a change to the table attributes/structure. Only applicable for Redshift tables; please see the Civis help desk for more info.
- refreshId
  string, The ID of the most recent statistics refresh.
- lastRun
  list, A list containing the following elements:
  - id integer,
  - state string,
  - createdAt string, The time that the run was queued.
  - startedAt string, The time that the run started.
  - finishedAt string, The time that the run completed.
  - error string, The error message for this run, if present.
- primaryKeys
  array, The primary keys for this table.
- lastModifiedKeys
  array, The columns indicating an entry’s modification status for this table.
- ontologyMapping
  list, The ontology-key to column-name mapping. See /ontology for the list of valid ontology keys.
- columns
  array, An array containing the following fields:
  - name string, Name of the column.
• civisDataType string, The generic data type of the column (ex. "string"). Since this is database-agnostic, it may be helpful when loading data to R/Python.
• sqlType string, The database-specific SQL type of the column (ex. "varchar(30)").
• sampleValues array, A sample of values from the column.
• encoding string, The compression encoding for this column See: http://docs.aws.amazon.com/redshift/latest/dg/c_Compression_encodings.html
• description string, The description of the column, as specified by the table owner
• order integer, Relative position of the column in the table.
• minValue string, Smallest value in the column.
• maxValue string, Largest value in the column.
• avgValue number, Average value of the column, where applicable.
• stddev number, Stddev of the column, where applicable.
• valueDistributionPercent object, A mapping between each value in the column and the percentage of rows with that value. Only present for tables with fewer than approximately 25,000,000 rows and for columns with fewer than twenty distinct values.
• coverageCount integer, Number of non-null values in the column.
• nullCount integer, Number of null values in the column.
• possibleDependentVariableTypes array, Possible dependent variable types the column may be used to model. Null if it may not be used as a dependent variable.
• useableAsIndependentVariable boolean, Whether the column may be used as an independent variable to train a model.
• useableAsPrimaryKey boolean, Whether the column may be used as a primary key to identify table rows.
• valueDistribution object, An object mapping distinct values in the column to the number of times they appear in the column.
• distinctCount integer, Number of distinct values in the column.

joins array, An array containing the following fields:
• id integer,
• leftTableId integer,
• leftIdentifier string,
• rightTableId integer,
• rightIdentifier string,
• on string,
• leftJoin boolean,
• createdAt string,
• updatedAt string,

multipartKey array,

enhancements array, An array containing the following fields:
• type string,
.tables_get_enhancements_cass_ncoa

- createdAt string,
- updatedAt string,
- joinId integer,

viewDef string,

tableDef string,

outgoingTableMatches array, An array containing the following fields:
- sourceTableId integer, Source table
- targetType string, Target type
- targetId integer, Target ID
- target object,
- job object,

.tables_get_enhancements_cass_ncoa

View the status of a CASS / NCOA table enhancement

Description

View the status of a CASS / NCOA table enhancement

Usage

tables_get_enhancements_cass_ncoa(id, source_table_id)

Arguments

id integer required. The ID of the enhancement.
source_table_id integer required. The ID of the table that was enhanced.

Value

A list containing the following elements:

id integer, The ID of the enhancement.
sourceTableId integer, The ID of the table that was enhanced.
state string, The state of the enhancement, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

enhancedTableSchema string, The schema name of the table created by the enhancement.
enhancedTableName string, The name of the table created by the enhancement.
performNcoa boolean, Whether to update addresses for records matching the National Change of Address (NCOA) database.
ncoaCredentialId
ingter, Credential to use when performing NCOA updates. Required if ‘performNcoa’ is true.

outputLevel
string, The set of fields persisted by a CASS or NCOA enhancement. For CASS enhancements, one of ‘cass’ or ‘all’. For NCOA enhancements, one of ‘cass’, ‘ncoa’, ‘coalesced’ or ‘all’. By default, all fields will be returned.

---

tables_get_enhancements_geocodings

View the status of a geocoding table enhancement

**Description**

View the status of a geocoding table enhancement

**Usage**

tables_get_enhancements_geocodings(id, source_table_id)

**Arguments**

id
integer required. The ID of the enhancement.

source_table_id
integer required. The ID of the table that was enhanced.

**Value**

A list containing the following elements:

id
integer, The ID of the enhancement.

sourceTableId
integer, The ID of the table that was enhanced.

state
string, The state of the enhancement, one of ‘queued’ ‘running’ ‘succeeded’ ‘failed’ or ‘cancelled’.

enhancedTableSchema
string, The schema name of the table created by the enhancement.

enhancedTableName
string, The name of the table created by the enhancement.
tables_list  List tables

Description
List tables

Usage

tables_list(
    database_id = NULL,
    schema = NULL,
    name = NULL,
    search = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)

Arguments
database_id  integer optional. The ID of the database.
schema  string optional. If specified, will be used to filter the tables returned. Substring matching is supported with "%" and "*" wildcards (e.g., "schema=%census%" will return both "client_census.table" and "census_2010.table").
name  string optional. If specified, will be used to filter the tables returned. Substring matching is supported with "%" and "*" wildcards (e.g., "name=%table%" will return both "table1" and "my table").
search  string optional. If specified, will be used to filter the tables returned. Will search across schema and name (in the full form schema.name) and will return any full name containing the search string.
limit  integer optional. Number of results to return. Defaults to 50. Maximum allowed is 1000.
page_num  integer optional. Page number of the results to return. Defaults to the first page, 1.
order  string optional. The field on which to order the result set. Defaults to schema. Must be one of: schema, name, search.
order_dir  string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value
An array containing the following fields:

id  integer, The ID of the table.
**tables_list_columns**

List columns in the specified table

**Description**

List columns in the specified table

**Usage**

```python
tables_list_columns(
    id,
    name = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```
Arguments

id  integer required.
name  string optional. Search for columns with the given name, within the specified table.
limit  integer optional. Number of results to return. Defaults to its maximum of 50.
page_num  integer optional. Page number of the results to return. Defaults to the first page, 1.
order  string optional. The field on which to order the result set. Defaults to name. Must be one of: name, order.
order_dir  string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

name  string, Name of the column.
civisDataType  string, The generic data type of the column (ex. "string"). Since this is database-agnostic, it may be helpful when loading data to R/Python.
sqlType  string, The database-specific SQL type of the column (ex. "varchar(30)").
sampleValues  array, A sample of values from the column.
encoding  string, The compression encoding for this columnSee: http://docs.aws.amazon.com/redshift/latest/dg/c_Compression_encodings.html
description  string, The description of the column, as specified by the table owner
order  integer, Relative position of the column in the table.
minValue  string, Smallest value in the column.
maxValue  string, Largest value in the column.
avgValue  number, Average value of the column, where applicable.
stddev  number, Stddev of the column, where applicable.
valueDistributionPercent  list, A mapping between each value in the column and the percentage of rows with that value.Only present for tables with fewer than approximately 25,000,000 rows and for columns with fewer than twenty distinct values.
coverageCount  integer, Number of non-null values in the column.
nullCount  integer, Number of null values in the column.
possibleDependentVariableTypes  array, Possible dependent variable types the column may be used to model. Null if it may not be used as a dependent variable.
useableAsIndependentVariable  boolean, Whether the column may be used as an independent variable to train a model.
useableAsPrimaryKey  boolean, Whether the column may be used as an primary key to identify table rows.
valueDistribution
list. An object mapping distinct values in the column to the number of times they appear in the column.
distinctCount
integer. Number of distinct values in the column.

tables_list_projects  List the projects a Table belongs to

Description
List the projects a Table belongs to

Usage
tables_list_projects(id, hidden = NULL)

Arguments
id
integer required. The ID of the Table.
hidden
boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value
An array containing the following fields:
id
integer, The ID for this project.
author
list. A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
name
string, The name of this project.
description
string, A description of the project.
users
array. An array containing the following fields:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
autoShare
boolean,
createdAt
string,
updatedAt
string,
archived
string, The archival status of the requested item(s).
tables_patch | Update a table

**Description**
Update a table

**Usage**

tables_patch(
    id,
    ontology_mapping = NULL,
    description = NULL,
    primary_keys = NULL,
    last_modified_keys = NULL
)

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of the table.</td>
</tr>
<tr>
<td>ontology_mapping</td>
<td>list optional. The ontology-key to column-name mapping. See /ontology for the list of valid ontology keys.</td>
</tr>
<tr>
<td>description</td>
<td>string optional. The user-defined description of the table.</td>
</tr>
<tr>
<td>primary_keys</td>
<td>array optional. A list of column(s) which together uniquely identify a row in the data. These columns must not contain NULL values.</td>
</tr>
<tr>
<td>last_modified_keys</td>
<td>array optional. The columns indicating when a row was last modified.</td>
</tr>
</tbody>
</table>

**Value**

A list containing the following elements:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer, The ID of the table.</td>
</tr>
<tr>
<td>databaseId</td>
<td>integer, The ID of the database.</td>
</tr>
<tr>
<td>schema</td>
<td>string, The name of the schema containing the table.</td>
</tr>
<tr>
<td>name</td>
<td>string, Name of the table.</td>
</tr>
<tr>
<td>description</td>
<td>string, The description of the table, as specified by the table owner</td>
</tr>
<tr>
<td>isView</td>
<td>boolean, True if this table represents a view. False if it represents a regular table.</td>
</tr>
<tr>
<td>rowCount</td>
<td>integer, The number of rows in the table.</td>
</tr>
<tr>
<td>columnCount</td>
<td>integer, The number of columns in the table.</td>
</tr>
<tr>
<td>sizeMb</td>
<td>number, The size of the table in megabytes.</td>
</tr>
<tr>
<td>owner</td>
<td>string, The database username of the table’s owner.</td>
</tr>
</tbody>
</table>
distkey  string, The column used as the Amazon Redshift distkey.
sortkeys string, The column used as the Amazon Redshift sortkey.
refreshStatus string, How up-to-date the table’s statistics on row counts, null counts, distinct counts, and values distributions are. One of: refreshing, stale, or current.
lastRefresh string, The time of the last statistics refresh.
dataUpdatedAt string, The last time that Civis Platform captured a change in this table. Only applicable for Redshift tables; please see the Civis help desk for more info.
schemaUpdatedAt string, The last time that Civis Platform captured a change to the table attributes/structure. Only applicable for Redshift tables; please see the Civis help desk for more info.
refreshId string, The ID of the most recent statistics refresh.
lastRun list, A list containing the following elements:
  • id integer,
  • state string,
  • createdAt string, The time that the run was queued.
  • startedAt string, The time that the run started.
  • finishedAt string, The time that the run completed.
  • error string, The error message for this run, if present.
primaryKeys array, The primary keys for this table.
lastModifiedKeys array, The columns indicating an entry’s modification status for this table.
ontologyMapping list, The ontology-key to column-name mapping. See /ontology for the list of valid ontology keys.

---

tables_post_enhancements_cass_ncoa

**Standardize addresses in a table**

**Description**

Standardize addresses in a table

**Usage**

```python
tables_post_enhancements_cass_ncoa(
    source_table_id,
    perform_ncoa = NULL,
    ncoa_credential_id = NULL,
    output_level = NULL
)
```
**Arguments**

source_table_id
- integer required. The ID of the table to be enhanced.

perform_ncoa
- boolean optional. Whether to update addresses for records matching the National Change of Address (NCOA) database.

ncoa_credential_id
- integer optional. Credential to use when performing NCOA updates. Required if 'performNcoa' is true.

output_level
- string optional. The set of fields persisted by a CASS or NCOA enhancement. For CASS enhancements, one of 'cass' or 'all'. For NCOA enhancements, one of 'cass', 'ncoa', 'coalesced' or 'all'. By default, all fields will be returned.

**Value**

A list containing the following elements:

- `id` integer, The ID of the enhancement.
- `sourceTableId` integer, The ID of the table that was enhanced.
- `state` string, The state of the enhancement, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.
- `enhancedTableSchema` string, The schema name of the table created by the enhancement.
- `enhancedTableName` string, The name of the table created by the enhancement.
- `performNcoa` boolean, Whether to update addresses for records matching the National Change of Address (NCOA) database.
- `ncoaCredentialId` integer, Credential to use when performing NCOA updates. Required if 'performNcoa' is true.
- `outputLevel` string, The set of fields persisted by a CASS or NCOA enhancement. For CASS enhancements, one of 'cass' or 'all'. For NCOA enhancements, one of 'cass', 'ncoa', 'coalesced' or 'all'. By default, all fields will be returned.

---

tables_post_enhancements_geocodings

*Geocode a table*

**Description**

Geocode a table

**Usage**

`tables_post_enhancements_geocodings(source_table_id)`
tables_post_refresh

Arguments

source_table_id
integer required. The ID of the table to be enhanced.

Value

A list containing the following elements:

id integer, The ID of the enhancement.
sourceTableId integer, The ID of the table that was enhanced.
state string, The state of the enhancement, one of 'queued' 'running' 'succeeded' 'failed' or 'cancelled'.

enhancedTableSchema string, The schema name of the table created by the enhancement.
enhancedTableName string, The name of the table created by the enhancement.

Description

Request a refresh for column and table statistics

Usage

tables_post_refresh(id)

Arguments

id integer required.

Value

A list containing the following elements:

id integer, The ID of the table.
databaseId integer, The ID of the database.
schema string, The name of the schema containing the table.
name string, Name of the table.
description string, The description of the table, as specified by the table owner
isView boolean, True if this table represents a view. False if it represents a regular table.
rowCount integer, The number of rows in the table.
columnCount integer, The number of columns in the table.
sizeMb number, The size of the table in megabytes.
owner string, The database username of the table’s owner.
distkey string, The column used as the Amazon Redshift distkey.
sortkeys string, The column used as the Amazon Redshift sortkey.
refreshStatus string, How up-to-date the table’s statistics on row counts, null counts, distinct counts, and values distributions are. One of: refreshing, stale, or current.
lastRefresh string, The time of the last statistics refresh.
dataUpdatedAt string, The last time that Civis Platform captured a change in this table. Only applicable for Redshift tables; please see the Civis help desk for more info.
schemaUpdatedAt string, The last time that Civis Platform captured a change to the table attributes/structure. Only applicable for Redshift tables; please see the Civis help desk for more info.
refreshId string, The ID of the most recent statistics refresh.
lastRun list, A list containing the following elements:
  • id integer,
  • state string,
  • createdAt string, The time that the run was queued.
  • startedAt string, The time that the run started.
  • finishedAt string, The time that the run completed.
  • error string, The error message for this run, if present.
primaryKeys array, The primary keys for this table.
lastModifiedKeys array, The columns indicating an entry’s modification status for this table.
ontologyMapping list, The ontology-key to column-name mapping. See /ontology for the list of valid ontology keys.
columns array, An array containing the following fields:
  • name string, Name of the column.
  • civisDataType string, The generic data type of the column (ex. "string"). Since this is database-agnostic, it may be helpful when loading data to R/Python.
  • sqlType string, The database-specific SQL type of the column (ex. "varchar(30)").
  • sampleValues array, A sample of values from the column.
  • encoding string, The compression encoding for this column See: http://docs.aws.amazon.com/redshift/
  • description string, The description of the column, as specified by the table owner
  • order integer, Relative position of the column in the table.
  • minValue string, Smallest value in the column.
  • maxValue string, Largest value in the column.
  • avgValue number, Average value of the column, where applicable.
  • stddev number, Stddev of the column, where applicable.
• valueDistributionPercent object, A mapping between each value in the column and the percentage of rows with that value. Only present for tables with fewer than approximately 25,000,000 rows and for columns with fewer than twenty distinct values.
• coverageCount integer, Number of non-null values in the column.
• nullCount integer, Number of null values in the column.
• possibleDependentVariableTypes array, Possible dependent variable types the column may be used to model. Null if it may not be used as a dependent variable.
• useableAsIndependentVariable boolean, Whether the column may be used as an independent variable to train a model.
• useableAsPrimaryKey boolean, Whether the column may be used as an primary key to identify table rows.
• valueDistribution object, An object mapping distinct values in the column to the number of times they appear in the column.
• distinctCount integer, Number of distinct values in the column.

joins array, An array containing the following fields:
• id integer,
• leftTableId integer,
• leftIdentifier string,
• rightTableId integer,
• rightIdentifier string,
• on string,
• leftJoin boolean,
• createdAt string,
• updatedAt string,

duplicates array,

enhancements array, An array containing the following fields:
• type string,
• createdAt string,
• updatedAt string,
• joinId integer,

viewDef string,

tableDef string,

outgoingTableMatches array, An array containing the following fields:
• sourceTableId integer, Source table
• targetType string, Target type
• targetId integer, Target ID
• target object,
• job object,
### tables_post_scan

**Description**

Creates and enqueues a single table scanner job on a new table.

**Usage**

```python
tables_post_scan(database_id, schema, table_name, stats_priority = NULL)
```

**Arguments**

- **database_id**: integer required. The ID of the database.
- **schema**: string required. The name of the schema containing the table.
- **table_name**: string required. The name of the table.
- **stats_priority**: string optional. When to sync table statistics. Valid Options are the following. Option: 'flag' means to flag stats for the next scheduled run of a full table scan on the database. Option: 'block' means to block this job on stats syncing. Option: 'queue' means to queue a separate job for syncing stats and do not block this job on the queued job. Defaults to 'flag'.

**Value**

A list containing the following elements:

- **jobId**: integer, The ID of the job created.
- **runId**: integer, The ID of the run created.

### tables_put_projects

**Description**

Add a Table to a project.

**Usage**

```python
tables_put_projects(id, project_id)
```

**Arguments**

- **id**: integer required. The ID of the Table.
- **project_id**: integer required. The ID of the project.
templates_delete_reports

Archive a Report Template (deprecated, use archiving endpoints instead)

Value
An empty HTTP response

templates_delete_reports_shares_groups

Revoke the permissions a group has on this object

Description
Revoke the permissions a group has on this object

Usage
templates_delete_reports_shares_groups(id, group_id)

Arguments
id integer required. The ID of the resource that is shared.
group_id integer required. The ID of the group.

Value
An empty HTTP response
templates_delete_reports_shares_users

Revoke the permissions a user has on this object

Description
Revoke the permissions a user has on this object

Usage
templates_delete_reports_shares_users(id, user_id)

Arguments
id integer required. The ID of the resource that is shared.
user_id integer required. The ID of the user.

Value
An empty HTTP response

templates_delete_scripts

Archive a Script Template (deprecated, use archiving endpoints instead)

Description
Archive a Script Template (deprecated, use archiving endpoints instead)

Usage
templates_delete_scripts(id)

Arguments
id integer required.

Value
An empty HTTP response
templates_delete_scripts_projects

Remove a Script Template from a project

Description

Remove a Script Template from a project

Usage

templates_delete_scripts_projects(id, project_id)

Arguments

id integer required. The ID of the Script Template.
project_id integer required. The ID of the project.

Value

An empty HTTP response

templates_delete_scripts_shares_groups

Revoke the permissions a group has on this object

Description

Revoke the permissions a group has on this object

Usage

templates_delete_scripts_shares_groups(id, group_id)

Arguments

id integer required. The ID of the resource that is shared.
group_id integer required. The ID of the group.

Value

An empty HTTP response
templates_delete_scriptsShares_users

*Revoke the permissions a user has on this object*

**Description**
Revoke the permissions a user has on this object

**Usage**
```
templates_delete_scriptsShares_users(id, user_id)
```

**Arguments**
- **id** integer required. The ID of the resource that is shared.
- **user_id** integer required. The ID of the user.

**Value**
An empty HTTP response

---

templates_get_reports  *Get a Report Template*

**Description**
Get a Report Template

**Usage**
```
templates_get_reports(id)
```

**Arguments**
- **id** integer required.

**Value**
A list containing the following elements:
- **id** integer.
- **name** string, The name of the template.
- **category** string, The category of this report template. Can be left blank. Acceptable values are: dataset-viz
- **createdAt** string,
templates_get_scripts

Get a Script Template

Description

Get a Script Template

Usage

templates_get_scripts(id)

Arguments

id integer required.

Value

A list containing the following elements:

id integer,
public boolean, If the template is public or not.
scriptId integer, The id of the script that this template uses.
scriptType string, The type of the template’s backing script (e.g SQL, Container, Python, R, JavaScript)
userContext string, The user context of the script that this template uses.
name string, The name of the template.
category string, The category of this template.
note string, A note describing what this template is used for; custom scripts created off this template will display this description.
createdAt string,
updatedAt string,
useCount integer, The number of uses of this template.
uReportId integer, The id of the report that this template uses.
technoReviewed boolean, Whether this template has been audited by Civis for security vulnerability and correctness.
archived boolean, Whether the template has been archived.
hidden boolean, The hidden status of the item.

---

templates_list_reports

*List Report Templates*

**Description**

List Report Templates

**Usage**

```python
templates_list_reports(
    hidden = NULL,
    category = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```

**Arguments**

- **hidden** boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
- **category** string optional. A category to filter results by, one of: dataset-viz
- **limit** integer optional. Number of results to return. Defaults to 50. Maximum allowed is 1000.
- **page_num** integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order** string optional. The field on which to order the result set. Defaults to name. Must be one of: name, updated_at, created_at.
- **order_dir** string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.
Value

An array containing the following fields:

- **id** integer,
- **name** string, The name of the template.
- **category** string, The category of this report template. Can be left blank. Acceptable values are: dataset-viz
- **createdAt** string,
- **updatedAt** string,
- **useCount** integer, The number of uses of this template.
- **archived** boolean, Whether the template has been archived.
- **author** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user's name.
  - **username** string, This user's username.
  - **initials** string, This user's initials.
  - **online** boolean, Whether this user is online.
- **techReviewed** boolean, Whether this template has been audited by Civis for security vulnerability and correctness.

**templates_list_reports_shares**

*List users and groups permissioned on this object*

Description

List users and groups permissioned on this object

Usage

`templates_list_reports_shares(id)`

Arguments

- **id** integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

- **readers** list, A list containing the following elements:
  - **users** array,
  - **groups** array,
writers list, A list containing the following elements:
• users array,
• groups array,

owners list, A list containing the following elements:
• users array,
• groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**templates_list_scripts**

*List Script Templates*

**Description**

List Script Templates

**Usage**

```r
templates_list_scripts(
  hidden = NULL,
  category = NULL,
  limit = NULL,
  page_num = NULL,
  order = NULL,
  order_dir = NULL
)
```

**Arguments**

- **hidden** boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
- **category** string optional. A category to filter results by, one of: import, export, enhancement, model, and script
- **limit** integer optional. Number of results to return. Defaults to 50. Maximum allowed is 1000.
- **page_num** integer optional. Page number of the results to return. Defaults to the first page, 1.
- **order** string optional. The field on which to order the result set. Defaults to name. Must be one of: name, updated_at, created_at.
- **order_dir** string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.
Value
An array containing the following fields:

- **id** integer,
- **public** boolean, If the template is public or not.
- **scriptId** integer, The id of the script that this template uses.
- **userContext** string, The user context of the script that this template uses.
- **name** string, The name of the template.
- **category** string, The category of this template.
- **createdAt** string,
- **updatedAt** string,
- **useCount** integer, The number of uses of this template.
- **uiReportId** integer, The id of the report that this template uses.
- **techReviewed** boolean, Whether this template has been audited by Civis for security vulnerability and correctness.
- **archived** boolean, Whether the template has been archived.

templates_list_scripts_projects
List the projects a Script Template belongs to

Description
List the projects a Script Template belongs to

Usage
templates_list_scripts_projects(id, hidden = NULL)

Arguments

- **id** integer required. The ID of the Script Template.
- **hidden** boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value
An array containing the following fields:

- **id** integer, The ID for this project.
- **author** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
templates_list_scripts_shares

• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

name string, The name of this project.
description string, A description of the project.
users array, An array containing the following fields:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

autoShare boolean,
createdAt string,
updatedAt string,
archived string, The archival status of the requested item(s).

templates_list_scripts_shares

List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

templates_list_scripts_shares(id)

Arguments

id integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers list, A list containing the following elements:
  • users array,
  • groups array,

writers list, A list containing the following elements:
  • users array,
  • groups array,

owners list, A list containing the following elements:
templates_patch_reports

- users array,
- groups array,

totalUserShares

  integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares

  integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

templates_patch_reports

  Update some attributes of this Report Template

Description

Update some attributes of this Report Template

Usage

  templates_patch_reports(
      id,
      name = NULL,
      category = NULL,
      archived = NULL,
      code_body = NULL,
      provide_api_key = NULL
  )

Arguments

  id
    integer required.

  name
    string optional. The name of the template.

  category
    string optional. The category of this report template. Can be left blank. Acceptable values are: dataset-viz

  archived
    boolean optional. Whether the template has been archived.

  code_body
    string optional. The code for the Template body.

  provide_api_key
    boolean optional. Whether reports based on this template request an API Key from the report viewer.
Value

A list containing the following elements:

- **id** integer,
- **name** string, The name of the template.
- **category** string, The category of this report template. Can be left blank. Acceptable values are: dataset-viz
- **createdAt** string,
- **updatedAt** string,
- **useCount** integer, The number of uses of this template.
- **archived** boolean, Whether the template has been archived.
- **author** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **techReviewed** boolean, Whether this template has been audited by Civis for security vulnerability and correctness.
- **authCodeUrl** string, A URL to the template’s stored code body.
- **provideAPIKey** boolean, Whether reports based on this template request an API Key from the report viewer.
- **hidden** boolean, The hidden status of the item.

---

`templates_patch_scripts`

*Update some attributes of this Script Template*

---

Description

Update some attributes of this Script Template

Usage

```python
templates_patch_scripts(
    id,
    name = NULL,
    note = NULL,
    ui_report_id = NULL,
    archived = NULL
)
```
templates_post_reports

Create a Report Template

Description

Create a Report Template

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td>Required. The name of the template.</td>
</tr>
<tr>
<td>name</td>
<td>string</td>
<td>Optional. The name of the template.</td>
</tr>
<tr>
<td>note</td>
<td>string</td>
<td>Optional. A note describing what this template is used for; custom scripts</td>
</tr>
<tr>
<td></td>
<td></td>
<td>created off this template will display this description.</td>
</tr>
<tr>
<td>ui_report_id</td>
<td>integer</td>
<td>Optional. The id of the report that this template uses.</td>
</tr>
<tr>
<td>archived</td>
<td>boolean</td>
<td>Optional. Whether the template has been archived.</td>
</tr>
</tbody>
</table>

Value

A list containing the following elements:

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td></td>
</tr>
<tr>
<td>public</td>
<td>boolean</td>
<td>If the template is public or not.</td>
</tr>
<tr>
<td>scriptId</td>
<td>integer</td>
<td>The id of the script that this template uses.</td>
</tr>
<tr>
<td>scriptType</td>
<td>string</td>
<td>The type of the template’s backing script (e.g SQL, Container, Python, R,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>JavaScript)</td>
</tr>
<tr>
<td>userContext</td>
<td>string</td>
<td>The user context of the script that this template uses.</td>
</tr>
<tr>
<td>name</td>
<td>string</td>
<td>The name of the template.</td>
</tr>
<tr>
<td>category</td>
<td>string</td>
<td>The category of this template.</td>
</tr>
<tr>
<td>note</td>
<td>string</td>
<td>A note describing what this template is used for; custom scripts created off</td>
</tr>
<tr>
<td></td>
<td></td>
<td>this template will display this description.</td>
</tr>
<tr>
<td>createdAt</td>
<td>string</td>
<td></td>
</tr>
<tr>
<td>updatedAt</td>
<td>string</td>
<td></td>
</tr>
<tr>
<td>useCount</td>
<td>integer</td>
<td>The number of uses of this template.</td>
</tr>
<tr>
<td>uiReportId</td>
<td>integer</td>
<td>The id of the report that this template uses.</td>
</tr>
<tr>
<td>techReviewed</td>
<td>boolean</td>
<td>Whether this template has been audited by Civis for security vulnerability</td>
</tr>
<tr>
<td></td>
<td></td>
<td>and correctness.</td>
</tr>
<tr>
<td>archived</td>
<td>boolean</td>
<td>Whether the template has been archived.</td>
</tr>
<tr>
<td>hidden</td>
<td>boolean</td>
<td>The hidden status of the item.</td>
</tr>
</tbody>
</table>
Usage

```python
templates_post_reports(
    name,
    code_body,
    category = NULL,
    archived = NULL,
    provide_api_key = NULL,
    hidden = NULL
)
```

Arguments

- **name** string required. The name of the template.
- **code_body** string required. The code for the Template body.
- **category** string optional. The category of this report template. Can be left blank. Acceptable values are: dataset-viz
- **archived** boolean optional. Whether the template has been archived.
- **provide_api_key** boolean optional. Whether reports based on this template request an API Key from the report viewer.
- **hidden** boolean optional. The hidden status of the item.

Value

A list containing the following elements:

- **id** integer,
- **name** string, The name of the template.
- **category** string, The category of this report template. Can be left blank. Acceptable values are: dataset-viz
- **createdAt** string,
- **updatedAt** string,
- **useCount** integer, The number of uses of this template.
- **archived** boolean, Whether the template has been archived.
- **author** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **techReviewed** boolean, Whether this template has been audited by Civis for security vulnerability and correctness.
- **authCodeUrl** string, A URL to the template’s stored code body.
**templates_post_reports_review**

Review a template for security vulnerability and correctness (admin-only)

**Description**

Review a template for security vulnerability and correctness (admin-only)

**Usage**

`templates_post_reports_review(id, status)`

**Arguments**

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The ID of the item.</td>
</tr>
<tr>
<td>status</td>
<td>boolean required. Whether this item has been reviewed.</td>
</tr>
</tbody>
</table>

**Value**

A list containing the following elements:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer,</td>
</tr>
<tr>
<td>name</td>
<td>string, The name of the template.</td>
</tr>
<tr>
<td>category</td>
<td>string, The category of this report template. Can be left blank. Acceptable values are: dataset-viz</td>
</tr>
<tr>
<td>createdAt</td>
<td>string,</td>
</tr>
<tr>
<td>updatedAt</td>
<td>string,</td>
</tr>
<tr>
<td>useCount</td>
<td>integer, The number of uses of this template.</td>
</tr>
<tr>
<td>archived</td>
<td>boolean, Whether the template has been archived.</td>
</tr>
<tr>
<td>author</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• id integer, The ID of this user.</td>
</tr>
<tr>
<td></td>
<td>• name string, This user’s name.</td>
</tr>
<tr>
<td></td>
<td>• username string, This user’s username.</td>
</tr>
<tr>
<td></td>
<td>• initials string, This user’s initials.</td>
</tr>
<tr>
<td></td>
<td>• online boolean, Whether this user is online.</td>
</tr>
<tr>
<td>techReviewed</td>
<td>boolean, Whether this template has been audited by Civis for security vulnerability and correctness.</td>
</tr>
<tr>
<td>authCodeUrl</td>
<td>string, A URL to the template’s stored code body.</td>
</tr>
<tr>
<td>provideAPIKey</td>
<td>boolean, Whether reports based on this template request an API Key from the report viewer.</td>
</tr>
<tr>
<td>hidden</td>
<td>boolean, The hidden status of the item.</td>
</tr>
</tbody>
</table>
templates_post_scripts

Create a Script Template

Description

Create a Script Template

Usage

```r
templates_post_scripts(
  script_id,
  name,
  note = NULL,
  ui_report_id = NULL,
  archived = NULL,
  hidden = NULL
)
```

Arguments

- `script_id` integer required. The id of the script that this template uses.
- `name` string required. The name of the template.
- `note` string optional. A note describing what this template is used for; custom scripts created off this template will display this description.
- `ui_report_id` integer optional. The id of the report that this template uses.
- `archived` boolean optional. Whether the template has been archived.
- `hidden` boolean optional. The hidden status of the item.

Value

A list containing the following elements:

- `id` integer,
- `public` boolean, If the template is public or not.
- `scriptId` integer, The id of the script that this template uses.
- `scriptIdType` string, The type of the template’s backing script (e.g SQL, Container, Python, R, JavaScript)
- `userContext` string, The user context of the script that this template uses.
- `name` string, The name of the template.
- `category` string, The category of this template.
- `note` string, A note describing what this template is used for; custom scripts created off this template will display this description.
templates_post_scripts_review

Review a template for security vulnerability and correctness (admin-only)

Description

Review a template for security vulnerability and correctness (admin-only)

Usage

templates_post_scripts_review(id, status)

Arguments

id integer required. The ID of the item.
status boolean required. Whether this item has been reviewed.

Value

A list containing the following elements:

id integer,
public boolean, If the template is public or not.
scriptId integer, The id of the script that this template uses.
scriptType string, The type of the template’s backing script (e.g SQL, Container, Python, R, JavaScript)
userContext string, The user context of the script that this template uses.
name string, The name of the template.
category string, The category of this template.
note string, A note describing what this template is used for; custom scripts created off this template will display this description.
createdAt string,
updatedAt string,
useCount integer, The number of uses of this template.
uiReportId integer, The id of the report that this template uses.
technicalReviewed boolean, Whether this template has been audited by Civis for security vulnerability and correctness.
archived boolean, Whether the template has been archived.
hidden boolean, The hidden status of the item.

---

**templates_put_reports**  
Replace all attributes of this Report Template

---

**Description**

Replace all attributes of this Report Template

**Usage**

```python
templates_put_reports(
    id,
    name,
    code_body,
    category = NULL,
    archived = NULL,
    provide_api_key = NULL
)
```

**Arguments**

- **id** integer required.
- **name** string required. The name of the template.
- **code_body** string required. The code for the Template body.
- **category** string optional. The category of this report template. Can be left blank. Acceptable values are: dataset-viz
- **archived** boolean optional. Whether the template has been archived.
- **provide_api_key** boolean optional. Whether reports based on this template request an API Key from the report viewer.

**Value**

A list containing the following elements:

- **id** integer,
- **name** string, The name of the template.
- **category** string, The category of this report template. Can be left blank. Acceptable values are: dataset-viz
createdAt string,
updatedAt string,
useCount integer, The number of uses of this template.
archived boolean, Whether the template has been archived.
author list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
techReviewed boolean, Whether this template has been audited by Civis for security vulnerability and correctness.
authCodeUrl string, A URL to the template’s stored code body.
provideAPIKey boolean, Whether reports based on this template request an API Key from the report viewer.
hidden boolean, The hidden status of the item.

templates_put_reports_shares_groups

Set the permissions groups has on this object

Description

Set the permissions groups has on this object

Usage

templates_put_reports_shares_groups(  
id,  
  group_ids,  
  permission_level,  
  share_email_body = NULL,  
  send_shared_email = NULL  
)

Arguments

id integer required. The ID of the resource that is shared.
group_ids array required. An array of one or more group IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.
templates_put_reports_shares_users

Value

A list containing the following elements:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,

- **writers** list, A list containing the following elements:
  - users array,
  - groups array,

- **owners** list, A list containing the following elements:
  - users array,
  - groups array,

- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**templates_put_reports_shares_users**

*Set the permissions users have on this object*

Description

Set the permissions users have on this object

Usage

```python
templates_put_reports_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```

Arguments

- **id** integer required. The ID of the resource that is shared.
- **user_ids** array required. An array of one or more user IDs.
- **permission_level** string required. Options are: "read", "write", or "manage".
- **share_email_body** string optional. Custom body text for e-mail sent on a share.
- **send_shared_email** boolean optional. Send email to the recipients of a share.
**Value**

A list containing the following elements:

- **readers** list, A list containing the following elements:
  - users array,
  - groups array,

- **writers** list, A list containing the following elements:
  - users array,
  - groups array,

- **owners** list, A list containing the following elements:
  - users array,
  - groups array,

- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**templates_put_scripts**  
*Replace all attributes of this Script Template*

---

**Description**

Replace all attributes of this Script Template

**Usage**

```
templates_put_scripts(
    id,
    name,
    note = NULL,
    ui_report_id = NULL,
    archived = NULL
)
```

**Arguments**

- **id** integer required.
- **name** string required. The name of the template.
- **note** string optional. A note describing what this template is used for; custom scripts created off this template will display this description.
- **ui_report_id** integer optional. The id of the report that this template uses.
- **archived** boolean optional. Whether the template has been archived.
Value

A list containing the following elements:

- **id**: integer,
- **public**: boolean, If the template is public or not.
- **scriptId**: integer, The id of the script that this template uses.
- **scriptType**: string, The type of the template’s backing script (e.g SQL, Container, Python, R, JavaScript)
- **userContext**: string, The user context of the script that this template uses.
- **name**: string, The name of the template.
- **category**: string, The category of this template.
- **note**: string, A note describing what this template is used for; custom scripts created off this template will display this description.
- **createdAt**: string,
- **updatedAt**: string,
- **useCount**: integer, The number of uses of this template.
- **uiReportId**: integer, The id of the report that this template uses.
- **techReviewed**: boolean, Whether this template has been audited by Civis for security vulnerability and correctness.
- **archived**: boolean, Whether the template has been archived.
- **hidden**: boolean, The hidden status of the item.

templates_put_scripts_projects

*Add a Script Template to a project*

Description

Add a Script Template to a project

Usage

```
templates_put_scripts_projects(id, project_id)
```

Arguments

- **id**: integer required. The ID of the Script Template.
- **project_id**: integer required. The ID of the project.

Value

An empty HTTP response
templates_put_scripts_shares_groups

Set the permissions groups has on this object

Description
Set the permissions groups has on this object

Usage

templates_put_scripts_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)

Arguments

id integer required. The ID of the resource that is shared.
group_ids array required. An array of one or more group IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value

A list containing the following elements:

readers list, A list containing the following elements:
    • users array,
    • groups array,

writers list, A list containing the following elements:
    • users array,
    • groups array,

owners list, A list containing the following elements:
    • users array,
    • groups array,

totalUserShares integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares
integer. For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

templates_put_scripts_shares_users
Set the permissions users have on this object

Description
Set the permissions users have on this object

Usage
templates_put_scripts_shares_users(
    id,
    user_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)

Arguments
id integer required. The ID of the resource that is shared.
user_ids array required. An array of one or more user IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value
A list containing the following elements:
readers list, A list containing the following elements:
    • users array,
    • groups array,
writers list, A list containing the following elements:
    • users array,
    • groups array,
owners list, A list containing the following elements:
    • users array,
groups array,

- **totalUserShares** integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
- **totalGroupShares** integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

---

**transfer_table**

Transfer a table from one location to another.

**Description**

Transfer a table from one location to another.

**Usage**

```r
transfer_table(
  source_db,
  dest_db,
  source_table,
  dest_table,
  job_name = NULL,
  source_credential_id = NULL,
  dest_credential_id = NULL,
  interval = NULL,
  verbose = FALSE,
  advanced_options = NULL
)
```

**Arguments**

- **source_db** string or int, The name of the database where the source table is located. Optionally, could be the database ID.
- **dest_db** string or int, The name of the database where the table will be transferred. Optionally, could be the database ID.
- **source_table** string, Full name of the table to transfer, e.g., "schema.table".
- **dest_table** string, Full name of the table in the destination database, e.g., "schema.table".
- **job_name** string, optional, A name to give the job. If omitted, a random job name will be used.
- **source_credential_id** string or int, Optional credential ID for the source database. If NULL, the default credential will be used.
- **dest_credential_id** string or int, Optional credential ID for the source database. If NULL, the default credential will be used.
users_delete_api_keys

Revoke the specified API key

Description

Revoke the specified API key

Usage

users_delete_api_keys(id, key_id)

Arguments

id string, required. The ID of the user or 'me'.
key_id integer, required. The ID of the API key.

Value

A list containing the following elements:

id integer, The ID of the API key.
name string, The name of the API key.
expiresAt string, The date and time when the key expired.
createdAt string, The date and time when the key was created.
revokedAt string, The date and time when the key was revoked.

interval Number of seconds to wait between checks for job completion. If NULL, the default exponential backoff from await will be used.
verbose bool, Set to TRUE to print intermediate progress indicators.
advanced_options

A list of advanced options for the sync. See imports_post_syncs for details.

Value

A civis_api object.

See Also

Other tables: get_table_id(), refresh_table()

Examples

```r
## Not run:
transfer_table(source_db='Cluster A', dest_db='Cluster B',
               source_table='schema.tbl', dest_table='schema.tbl')

## End(Not run)
```
users_delete_me_favorites

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>lastUsedAt</td>
<td>string, The date and time when the key was last used.</td>
</tr>
<tr>
<td>scopes</td>
<td>array, The scopes which the key is permissioned on.</td>
</tr>
<tr>
<td>useCount</td>
<td>integer, The number of times the key has been used.</td>
</tr>
<tr>
<td>expired</td>
<td>boolean, True if the key has expired.</td>
</tr>
<tr>
<td>active</td>
<td>boolean, True if the key has neither expired nor been revoked.</td>
</tr>
<tr>
<td>constraints</td>
<td>array, An array containing the following fields:</td>
</tr>
<tr>
<td></td>
<td>• constraint string, The path matcher of the constraint.</td>
</tr>
<tr>
<td></td>
<td>• constraintType string, The type of constraint (exact/prefix/regex/verb).</td>
</tr>
<tr>
<td></td>
<td>• getAllowed boolean, Whether the constraint allows GET requests.</td>
</tr>
<tr>
<td></td>
<td>• headAllowed boolean, Whether the constraint allows HEAD requests.</td>
</tr>
<tr>
<td></td>
<td>• postAllowed boolean, Whether the constraint allows POST requests.</td>
</tr>
<tr>
<td></td>
<td>• putAllowed boolean, Whether the constraint allows PUT requests.</td>
</tr>
<tr>
<td></td>
<td>• patchAllowed boolean, Whether the constraint allows PATCH requests.</td>
</tr>
<tr>
<td></td>
<td>• deleteAllowed boolean, Whether the constraint allows DELETE requests.</td>
</tr>
</tbody>
</table>

users_delete_me_favorites

Unfavorite an item

Description
Unfavorite an item

Usage
users_delete_me_favorites(id)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer required. The id of the favorite.</td>
</tr>
</tbody>
</table>

Value
An empty HTTP response
users_get  
*Show info about a user*

**Description**
Show info about a user

**Usage**
```plaintext
users_get(id)
```

**Arguments**
```
id: integer required. The ID of this user.
```

**Value**
A list containing the following elements:
```
id: integer, The ID of this user.
user: string, The username of this user.
name: string, The name of this user.
email: string, The email of this user.
active: boolean, The account status of this user.
primaryGroupId: integer, The ID of the primary group of this user.
groups: array, An array containing the following fields:
  • id: integer, The ID of this group.
  • name: string, The name of this group.
  • organizationId: integer, The organization associated with this group.
city: string, The city of this user.
state: string, The state of this user.
timeZone: string, The time zone of this user.
initials: string, The initials of this user.
department: string, The department of this user.
title: string, The title of this user.
githubUsername: string, The GitHub username of this user.
prefersSmsOtp: boolean, The preference for phone authorization of this user
vpnEnabled: boolean, The availability of vpn for this user.
ssoDisabled: boolean, The availability of SSO for this user.
otpRequiredForLogin: boolean, The two factor authentication requirement for this user.
```
users_get_api_keys

Show the specified API key

**Description**

Show the specified API key

**Usage**

users_get_api_keys(id, key_id)

**Arguments**

- id: string required. The ID of the user or ‘me’.
- key_id: integer required. The ID of the API key.

**Value**

A list containing the following elements:

- id: integer, The ID of the API key.
- name: string, The name of the API key.
- expiresAt: string, The date and time when the key expired.
- createdAt: string, The date and time when the key was created.
- revokedAt: string, The date and time when the key was revoked.
- lastUsedAt: string, The date and time when the key was last used.
- scopes: array, The scopes which the key is permissioned on.
- useCount: integer, The number of times the key has been used.

**Fields**

- exemptFromOrgSmsOtpDisabled: boolean, Whether the user has SMS OTP enabled on an individual level. This field does not matter if the org does not have SMS OTP disabled.
- smsOtpAllowed: boolean, Whether the user is allowed to receive two factor authentication codes via SMS.
- robot: boolean, Whether the user is a robot.
- phone: string, The phone number of this user.
- organizationSlug: string, The slug of the organization the user belongs to.
- organizationSSODisableCapable: boolean, The user’s organization’s ability to disable sso for their users.
- organizationLoginType: string, The user’s organization’s login type.
- organizationSmsOtpDisabled: boolean, Whether the user’s organization has SMS OTP disabled.
users_list

List users

Usage

users_list(
    feature_flag = NULL,
    account_status = NULL,
    query = NULL,
    group_id = NULL,
    organization_id = NULL,
    exclude_groups = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)

Arguments

feature_flag string optional. Return users that have a feature flag enabled.
account_status string optional. The account status by which to filter users. May be one of "active", "inactive", or "all".
query string optional. Return users who match the given query, based on name, user, and email.
group_id integer optional. The ID of the group by which to filter users. Cannot be present if organization_id is.
users_list_api_keys

organization_id
   integer optional. The ID of the organization by which to filter users. Cannot be present if group_id is.
exclude_groups
   boolean optional. Whether or to exclude users’ groups. Default: false.
limit
   integer optional. Number of results to return. Defaults to 20. Maximum allowed is 10000.
page_num
   integer optional. Page number of the results to return. Defaults to the first page, 1.
order
   string optional. The field on which to order the result set. Defaults to name. Must be one of: name, user.
order_dir
   string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to asc.

Value

An array containing the following fields:

id
   integer, The ID of this user.
user
   string, The username of this user.
name
   string, The name of this user.
email
   string, The email of this user.
active
   boolean, The account status of this user.
primaryGroupId
   integer, The ID of the primary group of this user.
groups
   array. An array containing the following fields:
   • id integer, The ID of this group.
   • name string, The name of this group.
   • organizationId integer, The organization associated with this group.
createdAt
   string, The date and time when the user was created.
currentSignInAt
   string, The date and time when the user’s current session began.

users_list_api_keys
   Show API keys belonging to the specified user

Description

Show API keys belonging to the specified user

Usage

users_list_api_keys(
   id,
   limit = NULL,
   page_num = NULL,
   order = NULL,
   order_dir = NULL
)
Arguments

id
string required. The ID of the user or 'me'.

limit
integer optional. Number of results to return. Defaults to its maximum of 50.

page_num
integer optional. Page number of the results to return. Defaults to the first page, 1.

order
string optional. The field on which to order the result set. Defaults to id. Must be one of: id.

order_dir
string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id
integer, The ID of the API key.

name
string, The name of the API key.

expiresAt
string, The date and time when the key expired.

createdAt
string, The date and time when the key was created.

revokedAt
string, The date and time when the key was revoked.

lastUsedAt
string, The date and time when the key was last used.

scopes
array, The scopes which the key is permissioned on.

useCount
integer, The number of times the key has been used.

expired
boolean, True if the key has expired.

active
boolean, True if the key has neither expired nor been revoked.

constraintCount
integer, The number of constraints on the created key

users_list_me
Show info about the logged-in user

Description

Show info about the logged-in user

Usage

users_list_me()
Value

A list containing the following elements:

- `id` integer, The ID of this user.
- `name` string, This user’s name.
- `email` string, This user’s email address.
- `username` string, This user’s username.
- `initials` string, This user’s initials.
- `lastCheckedAnnouncements` string, The date and time at which the user last checked their announcements.
- `featureFlags` list, The feature flag settings for this user.
- `roles` array, The roles this user has, listed by slug.
- `preferences` list, This user’s preferences.
- `customBranding` string, The branding of Platform for this user.
- `primaryGroupId` integer, The ID of the primary group of this user.
- `groups` array, An array containing the following fields:
  - `id` integer, The ID of this group.
  - `name` string, The name of this group.
  - `organizationId` integer, The organization associated with this group.
- `organizationName` string, The name of the organization the user belongs to.
- `organizationSlug` string, The slug of the organization the user belongs to.
- `organizationDefaultThemeId` integer, The ID of the organization’s default theme.
- `createdAt` string, The date and time when the user was created.
- `signInCount` integer, The number of times the user has signed in.
- `assumingRole` boolean, Whether the user is assuming a role or not.
- `assumingAdmin` boolean, Whether the user is assuming admin.
- `assumingAdminExpiration` string, When the user’s admin role is set to expire.

Description

List Favorites
Usage

users_list_me_favorites(
    object_id = NULL,
    object_type = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)

Arguments

  object_id           integer optional. The id of the object. If specified as a query parameter, must also specify object_type parameter.
  object_type         string optional. The type of the object that is favorited. Valid options: Project
  limit               integer optional. Number of results to return. Defaults to 50. Maximum allowed is 1000.
  page_num            integer optional. Page number of the results to return. Defaults to the first page, 1.
  order               string optional. The field on which to order the result set. Defaults to created_at. Must be one of: created_at, object_type, object_id.
  order_dir           string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

  id                   integer, The id of the favorite.
  objectId             integer, The id of the object. If specified as a query parameter, must also specify object_type parameter.
  objectType           string, The type of the object that is favorited. Valid options: Project
  objectName           string, The name of the object that is favorited.
  createdAt            string, The time this favorite was created.

users_list_me_ui  UI configuration for logged-in user

Description

UI configuration for logged-in user

Usage

users_list_me_ui()
Value

A list containing the following elements:

- **id** integer, The ID of this user.
- **navigationMenus** list, Navigation menus visible to this user.
- **userMenus** list, User profile menu items available to this user.
- **userType** list, A list containing the following elements:
  - **vendor** boolean, This attribute is deprecated
  - **media** boolean, True if user has access to the Media Optimizer job type.
  - **mainApp** string, The slug for the main app for an app-only user account.
  - **appCount** integer, Number of apps this user has access to.
  - **reportsOnly** boolean, True if user is a reports-only user.
  - **reportsCreator** boolean, True if this user is allowed to create HTML reports.
- **zendeskToken** string, JSON web token for this user’s Zendesk widget.

---

users_patch

Update info about a user (must be an admin or client user admin)

Description

Update info about a user (must be an admin or client user admin)

Usage

```python
users_patch(
    id,
    name = NULL,
    email = NULL,
    active = NULL,
    primary_group_id = NULL,
    city = NULL,
    state = NULL,
    time_zone = NULL,
    initials = NULL,
    department = NULL,
    title = NULL,
    prefers_sms_otp = NULL,
    group_ids = NULL,
    vpn_enabled = NULL,
    sso_disabled = NULL,
    otp_required_for_login = NULL,
    exempt_from_org_sms_otp_disabled = NULL,
    robot = NULL,
    phone = NULL,
    password = NULL
)
```
Arguments

id integer required. The ID of this user.
name string optional. The name of this user.
email string optional. The email of this user.
active boolean optional. The account status of this user.
primary_group_id integer optional. The ID of the primary group of this user.
city string optional. The city of this user.
state string optional. The state of this user.
time_zone string optional. The time zone of this user.
initials string optional. The initials of this user.
department string optional. The department of this user.
title string optional. The title of this user.
prefers_sms_otp boolean optional. The preference for phone authorization of this user.
group_ids array optional. An array of ids of all the groups this user is in.
vpn_enabled boolean optional. The availability of vpn for this user.
sso_disabled boolean optional. The availability of SSO for this user.
otp_required_for_login boolean optional. The two factor authentication requirement for this user.
exempt_from_org_sms_otp_disabled boolean optional. Whether the user has SMS OTP enabled on an individual level. This field does not matter if the org does not have SMS OTP disabled.
robot boolean optional. Whether the user is a robot.
phone string optional. The phone number of this user.
password string optional. The password of this user.

Value

A list containing the following elements:

id integer, The ID of this user.
user string, The username of this user.
name string, The name of this user.
email string, The email of this user.
active boolean, The account status of this user.
primaryGroupId integer, The ID of the primary group of this user.
groups array, An array containing the following fields:
  • id integer, The ID of this group.
  • name string, The name of this group.
  • organizationId integer, The organization associated with this group.
users_patch_me

Description

Update info about the logged-in user

Usage

users_patch_me(preferences = NULL, last_checked_announcements = NULL)
Arguments

preferences list optional. A list containing the following elements:

- appIndexOrderField string, Order field for the apps index pages.
- appIndexOrderDir string, Order direction for the apps index pages.
- resultIndexOrderField string, Order field for the results index page.
- resultIndexOrderDir string, Order direction for the results index page.
- resultIndexTypeFilter string, Type filter for the results index page.
- resultIndexAuthorFilter string, Author filter for the results index page.
- resultIndexArchivedFilter string, Archived filter for the results index page.
- importIndexOrderField string, Order field for the imports index page.
- importIndexOrderDir string, Order direction for the imports index page.
- importIndexTypeFilter string, Type filter for the imports index page.
- importIndexAuthorFilter string, Author filter for the imports index page.
- importIndexDestFilter string, Destination filter for the imports index page.
- importIndexStatusFilter string, Status filter for the imports index page.
- importIndexArchivedFilter string, Archived filter for the imports index page.
- exportIndexOrderField string, Order field for the exports index page.
- exportIndexOrderDir string, Order direction for the exports index page.
- exportIndexTypeFilter string, Type filter for the exports index page.
- exportIndexAuthorFilter string, Author filter for the exports index page.
- exportIndexStatusFilter string, Status filter for the exports index page.
- modelIndexOrderField string, Order field for the models index page.
- modelIndexOrderDir string, Order direction for the models index page.
- modelIndexAuthorFilter string, Author filter for the models index page.
- modelIndexStatusFilter string, Status filter for the models index page.
- modelIndexArchivedFilter string, Archived filter for the models index page.
- modelIndexThumbnailView string, Thumbnail view for the models index page.
- scriptIndexOrderField string, Order field for the scripts index page.
- scriptIndexOrderDir string, Order direction for the scripts index page.
- scriptIndexTypeFilter string, Type filter for the scripts index page.
- scriptIndexAuthorFilter string, Author filter for the scripts index page.
- scriptIndexStatusFilter string, Status filter for the scripts index page.
- scriptIndexArchivedFilter string, Archived filter for the scripts index page.
- projectIndexOrderField string, Order field for the projects index page.
- projectIndexOrderDir string, Order direction for the projects index page.
- projectIndexAuthorFilter string, Author filter for the projects index page.
- projectIndexArchivedFilter string, Archived filter for the projects index page.
- reportIndexThumbnailView string, Thumbnail view for the reports index page.
- projectDetailOrderField string, Order field for projects detail pages.
- projectDetailOrderDir string, Order direction for projects detail pages.
• projectDetailAuthorFilter string, Author filter for projects detail pages.
• projectDetailTypeFilter string, Type filter for projects detail pages.
• projectDetailArchivedFilter string, Archived filter for the projects detail pages.
• enhancementIndexOrderField string, Order field for the enhancements index page.
• enhancementIndexOrderDir string, Order direction for the enhancements index page.
• enhancementIndexAuthorFilter string, Author filter for the enhancements index page.
• enhancementIndexArchivedFilter string, Archived filter for the enhancements index page.
• preferredServerId integer, ID of preferred server.
• civisExploreSkipIntro boolean, Whether the user is shown steps for each exploration.
• registrationIndexOrderField string, Order field for the registrations index page.
• registrationIndexOrderDir string, Order direction for the registrations index page.
• registrationIndexStatusFilter string, Status filter for the registrations index page.
• upgradeRequested string, Whether a free trial upgrade has been requested.
• welcomeOrderField string, Order direction for the welcome page.
• welcomeOrderDir string, Order direction for the welcome page.
• welcomeAuthorFilter string, Status filter for the welcome page.
• welcomeStatusFilter string, Status filter for the welcome page.
• welcomeArchivedFilter string, Status filter for the welcome page.
• dataPaneWidth string, Width of the data pane when expanded.
• dataPaneCollapsed string, Whether the data pane is collapsed.
• notebookOrderField string, Order field for the notebooks page.
• notebookOrderDir string, Order direction for the notebooks page.
• notebookAuthorFilter string, Author filter for the notebooks page.
• notebookArchivedFilter string, Archived filter for the notebooks page.
• notebookStatusFilter string, Status filter for the notebooks page.
• workflowIndexOrderField string, Order field for the workflows page.
• workflowIndexOrderDir string, Order direction for the workflows page.
• workflowIndexAuthorFilter string, Author filter for the workflows page.
• workflowIndexArchivedFilter string, Archived filter for the workflows page.
• serviceOrderField string, Order field for the services page.
• serviceOrderDir string, Order direction for the services page.
• serviceAuthorFilter string, Author filter for the services page.
• serviceArchivedFilter string, Archived filter for the services page.

last_checked_announcements

string optional. The date and time at which the user last checked their announcements.
Value

A list containing the following elements:

- **id** integer, The ID of this user.
- **name** string, This user’s name.
- **email** string, This user’s email address.
- **username** string, This user’s username.
- **initials** string, This user’s initials.
- **lastCheckedAnnouncements** string, The date and time at which the user last checked their announcements.
- **featureFlags** list, The feature flag settings for this user.
- **roles** array, The roles this user has, listed by slug.
- **preferences** list, This user’s preferences.
- **customBranding** string, The branding of Platform for this user.
- **primaryGroupId** integer, The ID of the primary group of this user.
- **groups** array, An array containing the following fields:
  - **id** integer, The ID of this group.
  - **name** string, The name of this group.
  - **organizationId** integer, The organization associated with this group.
- **organizationName** string, The name of the organization the user belongs to.
- **organizationSlug** string, The slug of the organization the user belongs to.
- **organizationDefaultThemeId** integer, The ID of the organization’s default theme.
- **createdAt** string, The date and time when the user was created.
- **signInCount** integer, The number of times the user has signed in.
- **assumingRole** boolean, Whether the user is assuming a role or not.
- **assumingAdmin** boolean, Whether the user is assuming admin.
- **assumingAdminExpiration** string, When the user’s admin role is set to expire.
Create a new user (must be an admin or client user admin)

Usage

users_post(
    name,
    email,
    primary_group_id,
    user,
    active = NULL,
    city = NULL,
    state = NULL,
    time_zone = NULL,
    initials = NULL,
    department = NULL,
    title = NULL,
    prefers_sms_otp = NULL,
    group_ids = NULL,
    vpn_enabled = NULL,
    sso_disabled = NULL,
    otp_required_for_login = NULL,
    exempt_from_org_sms_otp_disabled = NULL,
    robot = NULL,
    send_email = NULL
)

Arguments

name  string required. The name of this user.
email string required. The email of this user.
primary_group_id  integer required. The ID of the primary group of this user.
user  string required. The username of this user.
active  boolean optional. The account status of this user.
city  string optional. The city of this user.
state  string optional. The state of this user.
time_zone  string optional. The time zone of this user.
initials  string optional. The initials of this user.
department  string optional. The department of this user.
users_post

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>title</td>
<td>string optional</td>
<td>The title of this user.</td>
</tr>
<tr>
<td>prefers_sms_otp</td>
<td>boolean optional</td>
<td>The preference for phone authorization of this user</td>
</tr>
<tr>
<td>group_ids</td>
<td>array optional</td>
<td>An array of ids of all the groups this user is in.</td>
</tr>
<tr>
<td>vpn_enabled</td>
<td>boolean optional</td>
<td>The availability of vpn for this user.</td>
</tr>
<tr>
<td>sso_disabled</td>
<td>boolean optional</td>
<td>The availability of SSO for this user.</td>
</tr>
<tr>
<td>otp_required_for_login</td>
<td>boolean optional</td>
<td>The two factor authentication requirement for this user.</td>
</tr>
<tr>
<td>exempt_from_org_sms_otp_disabled</td>
<td>boolean optional</td>
<td>Whether the user has SMS OTP enabled on an individual level. This field does not matter if the org does not have SMS OTP disabled.</td>
</tr>
<tr>
<td>robot</td>
<td>boolean optional</td>
<td>Whether the user is a robot.</td>
</tr>
<tr>
<td>send_email</td>
<td>boolean optional</td>
<td>Whether the user will receive a welcome email.</td>
</tr>
</tbody>
</table>

**Value**

A list containing the following elements:

- id          integer, The ID of this user.
- user        string, The username of this user.
- name        string, The name of this user.
- email       string, The email of this user.
- active      boolean, The account status of this user.
- primaryGroupId integer, The ID of the primary group of this user.
- groups      array, An array containing the following fields:
  - id integer, The ID of this group.
  - name string, The name of this group.
  - organizationId integer, The organization associated with this group.
- city        string, The city of this user.
- state       string, The state of this user.
- timeZone    string, The time zone of this user.
- initials   string, The initials of this user.
- department string, The department of this user.
- title       string, The title of this user.
- githubUsername string, The GitHub username of this user.
- prefersSmsOtp boolean, The preference for phone authorization of this user
- vpnEnabled boolean, The availability of vpn for this user.
- ssoDisabled boolean, The availability of SSO for this user.
- otpRequiredForLogin boolean, The two factor authentication requirement for this user.
exemptFromOrgSmsOtpDisabled
  boolean, Whether the user has SMS OTP enabled on an individual level. This field does not matter if the org does not have SMS OTP disabled.
smsOtpAllowed
  boolean, Whether the user is allowed to receive two factor authentication codes via SMS.
robot
  boolean, Whether the user is a robot.
phone
  string, The phone number of this user.
organizationSlug
  string, The slug of the organization the user belongs to.
organizationSSODisableCapable
  boolean, The user’s organization’s ability to disable sso for their users.
organizationLoginType
  string, The user’s organization’s login type.
organizationSmsOtpDisabled
  boolean, Whether the user’s organization has SMS OTP disabled.

users_post_api_keys

Create a new API key belonging to the logged-in user

Description

Create a new API key belonging to the logged-in user

Usage

users_post_api_keys(id, expires_in, name, constraints = NULL)

Arguments

  id
    string required. The ID of the user or `me`.
  expires_in
    integer required. The number of seconds the key should last for.
  name
    string required. The name of the API key.
  constraints
    array optional. An array containing the following fields:
      • constraint string, The path matcher of the constraint.
      • constraintType string, The type of constraint (exact/prefix/regex/verb).
      • getAllowed boolean, Whether the constraint allows GET requests.
      • headAllowed boolean, Whether the constraint allows HEAD requests.
      • postAllowed boolean, Whether the constraint allows POST requests.
      • putAllowed boolean, Whether the constraint allows PUT requests.
      • patchAllowed boolean, Whether the constraint allows PATCH requests.
      • deleteAllowed boolean, Whether the constraint allows DELETE requests.
Users Post Me Favorites

**Value**

A list containing the following elements:

- **id** integer, The ID of the API key.
- **name** string, The name of the API key.
- **expiresAt** string, The date and time when the key expired.
- **createdAt** string, The date and time when the key was created.
- **revokedAt** string, The date and time when the key was revoked.
- **lastUsedAt** string, The date and time when the key was last used.
- **scopes** array, The scopes which the key is permissioned on.
- **useCount** integer, The number of times the key has been used.
- **expired** boolean, True if the key has expired.
- **active** boolean, True if the key has neither expired nor been revoked.
- **constraints** array, An array containing the following fields:
  - **constraint** string, The path matcher of the constraint.
  - **constraintType** string, The type of constraint (exact/prefix/regex/verb).
  - **getAllowed** boolean, Whether the constraint allows GET requests.
  - **headAllowed** boolean, Whether the constraint allows HEAD requests.
  - **postAllowed** boolean, Whether the constraint allows POST requests.
  - **putAllowed** boolean, Whether the constraint allows PUT requests.
  - **patchAllowed** boolean, Whether the constraint allows PATCH requests.
  - **deleteAllowed** boolean, Whether the constraint allows DELETE requests.
- **token** string, The API key.

---

**Description**

Favorite an item

**Usage**

`users_post_me_favorites(object_id, object_type)`

**Arguments**

- **object_id** integer required. The id of the object. If specified as a query parameter, must also specify object_type parameter.
- **object_type** string required. The type of the object that is favorited. Valid options: Project
**workflows_delete_projects**

*Remove a Workflow from a project*

**Description**

Remove a Workflow from a project

**Usage**

`workflows_delete_projects(id, project_id)`

**Arguments**

- **id** integer required. The ID of the Workflow.
- **project_id** integer required. The ID of the project.

**Value**

An empty HTTP response

---

**workflows_delete_shares_groups**

*Revoke the permissions a group has on this object*

**Description**

Revoke the permissions a group has on this object

**Usage**

`workflows_delete_shares_groups(id, group_id)`
workflows_get

Arguments

id                  integer required. The ID of the resource that is shared.
group_id            integer required. The ID of the group.

Value

An empty HTTP response

workflows_delete_shares_users

Revoke the permissions a user has on this object

Description

Revoke the permissions a user has on this object

Usage

workflows_delete_shares_users(id, user_id)

Arguments

id                  integer required. The ID of the resource that is shared.
user_id             integer required. The ID of the user.

Value

An empty HTTP response

workflows_get

Get a Workflow

Description

Get a Workflow

Usage

workflows_get(id)

Arguments

id                  integer required.
Value

A list containing the following elements:

- **id** integer, The ID for this workflow.
- **name** string, The name of this workflow.
- **description** string, A description of the workflow.
- **definition** string, The definition of the workflow in YAML format. Must not be specified if ‘fromJobChain’ is specified.
- **valid** boolean, The validity of the workflow definition.
- **validationErrors** string, The errors encountered when validating the workflow definition.
- **fileId** string, The file id for the s3 file containing the workflow configuration.
- **user** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **state** string, The state of the workflow. State is “running” if any execution is running, otherwise reflects most recent execution state.
- **schedule** list, A list containing the following elements:
  - **scheduled** boolean, If the item is scheduled.
  - **scheduledDays** array, Day based on numeric value starting at 0 for Sunday.
  - **scheduledHours** array, Hours of the day it is scheduled on.
  - **scheduledMinutes** array, Minutes of the day it is scheduled on.
  - **scheduledRunsPerHour** integer, Alternative to scheduled minutes, number of times to run per hour.
- **allowConcurrentExecutions** boolean, Whether the workflow can execute when already running.
- **timeZone** string, The time zone of this workflow.
- **nextExecutionAt** string, The time of the next scheduled execution.
- **notifications** list, A list containing the following elements:
  - **urls** array, URLs to receive a POST request at job completion
  - **successEmailSubject** string, Custom subject line for success e-mail.
  - **successEmailBody** string, Custom body text for success e-mail, written in Markdown.
  - **successEmailAddresses** array, Addresses to notify by e-mail when the job completes successfully.
  - **failureEmailAddresses** array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on
• failureOn boolean, If failure email notifications are on

archived string, The archival status of the requested item(s).
hidden boolean, The hidden status of the item.
createdAt string,
updatedAt string,

---

**workflows_get_executions**

*Get a workflow execution*

---

**Description**

Get a workflow execution

**Usage**

`workflows_get_executions(id, execution_id)`

**Arguments**

- **id** integer required. The ID for the workflow.
- **execution_id** integer required. The ID for the workflow execution.

**Value**

A list containing the following elements:

- **id** integer, The ID for this workflow execution.
- **state** string, The state of this workflow execution.
- **mistralState** string, The state of this workflow as reported by mistral. One of running, paused, success, error, or cancelled
- **mistralStateInfo** string, The state info of this workflow as reported by mistral.
- **user** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **definition** string, The definition of the workflow for this execution.
input list, Key-value pairs defined for this execution.
includedTasks array, The subset of workflow tasks selected to execute.
tasks array, An array containing the following fields:
• name string, The name of the task.
• mistralState string, The state of this task. One of idle, waiting, running, delayed, success, error, or cancelled
• mistralStateInfo string, Extra info associated with the state of the task.
• runs array, The runs associated with this task, in descending order by id.
• executions array, The executions run by this task, in descending order by id.
startedAt string, The time this execution started.
finishedAt string, The time this execution finished.
createdAt string, The time this execution was created.
updatedAt string, The time this execution was last updated.

workflows_get_executions_tasks

Get a task of a workflow execution

Description

Get a task of a workflow execution

Usage

workflows_get_executions_tasks(id, execution_id, task_name)

Arguments

id integer required. The ID for the workflow.
execution_id integer required. The ID for the workflow execution.
task_name string required. The URL-encoded name of the task.

Value

A list containing the following elements:

name string, The name of the task.
mistralState string, The state of this task. One of idle, waiting, running, delayed, success, error, or cancelled
mistralStateInfo string, Extra info associated with the state of the task.
runs array, An array containing the following fields:
• id integer, The ID of the run.
• jobId integer, The ID of the job associated with the run.
• state string, The state of the run.
• createdAt string, The time that the run was queued.
• startedAt string, The time that the run started.
• finishedAt string, The time that the run completed.

executions array, An array containing the following fields:
• id integer, The ID of the execution.
• workflowId integer, The ID of the workflow associated with the execution.
• state string, The state of this workflow execution.
• createdAt string, The time this execution was created.
• startedAt string, The time this execution started.
• finishedAt string, The time this execution finished.

workflows_get_git_commits

Get file contents at commit_hash

Description

Get file contents at commit_hash

Usage

workflows_get_git_commits(id, commit_hash)

Arguments

id integer required. The ID of the file.
commit_hash string required. The SHA (full or shortened) of the desired git commit.

Value

A list containing the following elements:

content string, The file’s contents.
type string, The file’s type.
size integer, The file’s size.
fileHash string, The SHA of the file.
workflows_list

**Description**

List Workflows

**Usage**

```r
workflows_list(
    hidden = NULL,
    archived = NULL,
    author = NULL,
    state = NULL,
    scheduled = NULL,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```

**Arguments**

- `hidden` boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.
- `archived` string optional. The archival status of the requested item(s).
- `author` string optional. Author of the workflow. It accepts a comma-separated list of author ids.
- `state` array optional. State of the most recent execution. One or more of queued, running, succeeded, failed, cancelled, idle, and scheduled.
- `scheduled` boolean optional. If the workflow is scheduled.
- `limit` integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.
- `page_num` integer optional. Page number of the results to return. Defaults to the first page, 1.
- `order` string optional. The field on which to order the result set. Defaults to updated_at. Must be one of: updated_at, name, created_at.
- `order_dir` string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

**Value**

An array containing the following fields:

- `id` integer, The ID for this workflow.
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>name</td>
<td>string, The name of this workflow.</td>
</tr>
<tr>
<td>description</td>
<td>string, A description of the workflow.</td>
</tr>
<tr>
<td>valid</td>
<td>boolean, The validity of the workflow definition.</td>
</tr>
<tr>
<td>fileId</td>
<td>string, The file id for the s3 file containing the workflow configuration.</td>
</tr>
<tr>
<td>user</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• id integer, The ID of this user.</td>
</tr>
<tr>
<td></td>
<td>• name string, This user’s name.</td>
</tr>
<tr>
<td></td>
<td>• username string, This user’s username.</td>
</tr>
<tr>
<td></td>
<td>• initials string, This user’s initials.</td>
</tr>
<tr>
<td></td>
<td>• online boolean, Whether this user is online.</td>
</tr>
<tr>
<td>state</td>
<td>string, The state of the workflow. State is &quot;running&quot; if any execution is running, otherwise reflects most recent execution state.</td>
</tr>
<tr>
<td>schedule</td>
<td>list, A list containing the following elements:</td>
</tr>
<tr>
<td></td>
<td>• scheduled boolean, If the item is scheduled.</td>
</tr>
<tr>
<td></td>
<td>• scheduledDays array, Day based on numeric value starting at 0 for Sunday.</td>
</tr>
<tr>
<td></td>
<td>• scheduledHours array, Hours of the day it is scheduled on.</td>
</tr>
<tr>
<td></td>
<td>• scheduledMinutes array, Minutes of the day it is scheduled on.</td>
</tr>
<tr>
<td></td>
<td>• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.</td>
</tr>
<tr>
<td>allowConcurrentExecutions</td>
<td>boolean, Whether the workflow can execute when already running.</td>
</tr>
<tr>
<td>timeZone</td>
<td>string, The time zone of this workflow.</td>
</tr>
<tr>
<td>nextExecutionAt</td>
<td>string, The time of the next scheduled execution.</td>
</tr>
<tr>
<td>archived</td>
<td>string, The archival status of the requested item(s).</td>
</tr>
<tr>
<td>createdAt</td>
<td>string,</td>
</tr>
<tr>
<td>updatedAt</td>
<td>string,</td>
</tr>
</tbody>
</table>

**workflows_list_executions**

*List workflow executions*

**Description**

List workflow executions

**Usage**

```python
workflows_list_executions(
    id,
    limit = NULL,
    page_num = NULL,
    order = NULL,
    order_dir = NULL
)
```
Arguments

id
integer required. The ID for this workflow.

limit
integer optional. Number of results to return. Defaults to 20. Maximum allowed is 50.

page_num
integer optional. Page number of the results to return. Defaults to the first page, 1.

order
string optional. The field on which to order the result set. Defaults to id. Must be one of: id, updated_at, created_at.

order_dir
string optional. Direction in which to sort, either asc (ascending) or desc (descending) defaulting to desc.

Value

An array containing the following fields:

id
integer, The ID for this workflow execution.

state
string, The state of this workflow execution.

mistralState
string, The state of this workflow as reported by mistral. One of running, paused, success, error, or cancelled

mistralStateInfo
string, The state info of this workflow as reported by mistral.

user
list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.

startedAt
string, The time this execution started.

finishedAt
string, The time this execution finished.

createdAt
string, The time this execution was created.

updatedAt
string, The time this execution was last updated.

Description

Get the git metadata attached to an item

Usage

`workflows_list_git(id)`
**Arguments**

id integer required. The ID of the file.

**Value**

A list containing the following elements:

- **gitRef** string, A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
- **gitBranch** string, The git branch that the file is on.
- **gitPath** string, The path of the file in the repository.
- **gitRepo** list, A list containing the following elements:
  - **id** integer, The ID for this git repository.
  - **repoUrl** string, The URL for this git repository.
  - **createdAt** string,
  - **updatedAt** string,
- **pullFromGit** boolean, Automatically pull latest commit from git. Only works for scripts.

**Description**

Get the git commits for an item

**Usage**

`workflows_list_git_commits(id)`

**Arguments**

id integer required. The ID of the file.

**Value**

A list containing the following elements:

- **commitHash** string, The SHA of the commit.
- **authorName** string, The name of the commit’s author.
- **date** string, The commit’s timestamp.
- **message** string, The commit message.
workflows_list_projects

List the projects a Workflow belongs to

Description

List the projects a Workflow belongs to

Usage

workflows_list_projects(id, hidden = NULL)

Arguments

id integer required. The ID of the Workflow.
hidden boolean optional. If specified to be true, returns hidden items. Defaults to false, returning non-hidden items.

Value

An array containing the following fields:

- id integer, The ID for this project.
- author list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.
- name string, The name of this project.
- description string, A description of the project.
- users array, An array containing the following fields:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.
- autoShare boolean,
- createdAt string,
- updatedAt string,
- archived string, The archival status of the requested item(s).
workflows_list_shares  List users and groups permissioned on this object

Description

List users and groups permissioned on this object

Usage

workflows_list_shares(id)

Arguments

id  integer required. The ID of the resource that is shared.

Value

An array containing the following fields:

readers  list, A list containing the following elements:
  • users array,
  • groups array,

writers  list, A list containing the following elements:
  • users array,
  • groups array,

owners  list, A list containing the following elements:
  • users array,
  • groups array,

totalUserShares  integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.

totalGroupShares  integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.
workflows_patch

Update some attributes of this Workflow

Description

Update some attributes of this Workflow

Usage

workflows_patch(
    id,
    name = NULL,
    description = NULL,
    definition = NULL,
    schedule = NULL,
    allow_concurrent_executions = NULL,
    time_zone = NULL,
    notifications = NULL
)

Arguments

id integer required. The ID for this workflow.
name string optional. The name of this workflow.
description string optional. A description of the workflow.
definition string optional. The definition of the workflow in YAML format. Must not be specified if ‘fromJobChain’ is specified.
schedule list optional. A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.
allow_concurrent_executions boolean optional. Whether the workflow can execute when already running.
time_zone string optional. The time zone of this workflow.
notifications list optional. A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on
• failureOn boolean, If failure email notifications are on

Value

A list containing the following elements:

id integer, The ID for this workflow.
name string, The name of this workflow.
description string, A description of the workflow.
definition string, The definition of the workflow in YAML format. Must not be specified if ‘fromJobChain’ is specified.
valid boolean, The validity of the workflow definition.
validationErrors string, The errors encountered when validating the workflow definition.
fileId string, The file id for the s3 file containing the workflow configuration.
user list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
state string, The state of the workflow. State is "running" if any execution is running, otherwise reflects most recent execution state.
schedule list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.
allowConcurrentExecutions boolean, Whether the workflow can execute when already running.
timeZone string, The time zone of this workflow.
nextExecutionAt string, The time of the next scheduled execution.
notifications list, A list containing the following elements:
workflows_post

- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on
- failureOn boolean, If failure email notifications are on

archived string, The archival status of the requested item(s).
hidden boolean, The hidden status of the item.
createdAt string,
updatedAt string,

workflows_post  Create a Workflow

Description

Create a Workflow

Usage

workflows_post(
    name,
    description = NULL,
    from_job_chain = NULL,
    definition = NULL,
    schedule = NULL,
    allow_concurrent_executions = NULL,
    time_zone = NULL,
    notifications = NULL,
    hidden = NULL
)

Arguments

name string required. The name of this workflow.
description string optional. A description of the workflow.
from_job_chain integer optional. If specified, create a workflow from the job chain this job is in, and inherit the schedule from the root of the chain.
definition string optional. The definition of the workflow in YAML format. Must not be specified if ‘fromJobChain’ is specified.

schedule list optional. A list containing the following elements:
- scheduled boolean, If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

allow_concurrent_executions boolean optional. Whether the workflow can execute when already running.

time_zone string optional. The time zone of this workflow.

notifications list optional. A list containing the following elements:
- urls array, URLs to receive a POST request at job completion
- successEmailSubject string, Custom subject line for success e-mail.
- successEmailBody string, Custom body text for success e-mail, written in Markdown.
- successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
- failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
- stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
- successOn boolean, If success email notifications are on
- failureOn boolean, If failure email notifications are on

hidden boolean optional. The hidden status of the item.

Value
A list containing the following elements:

id integer, The ID for this workflow.

name string, The name of this workflow.

description string, A description of the workflow.

definition string, The definition of the workflow in YAML format. Must not be specified if ‘fromJobChain’ is specified.

valid boolean, The validity of the workflow definition.

validationErrors string, The errors encountered when validating the workflow definition.

fileId string, The file id for the s3 file containing the workflow configuration.

user list, A list containing the following elements:
- id integer, The ID of this user.
- name string, This user’s name.
• username string, This user's username.
• initials string, This user's initials.
• online boolean, Whether this user is online.

state string, The state of the workflow. State is "running" if any execution is running, otherwise reflects most recent execution state.

schedule list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

allowConcurrentExecutions boolean, Whether the workflow can execute when already running.

timeZone string, The time zone of this workflow.

nextExecutionAt string, The time of the next scheduled execution.

notifications list, A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  • successOn boolean, If success email notifications are on
  • failureOn boolean, If failure email notifications are on

archived string, The archival status of the requested item(s).

hidden boolean, The hidden status of the item.

createdAt string,

updatedAt string,
workflows_post_clone

Clone this Workflow

Description

Clone this Workflow

Usage

workflows_post_clone(id, clone_schedule = NULL, clone_notifications = NULL)

Arguments

id
  integer required. The ID for the workflow.
clone_schedule
  boolean optional. If true, also copy the schedule to the new workflow.
clone_notifications
  boolean optional. If true, also copy the notifications to the new workflow.

Value

A list containing the following elements:

id
  integer, The ID for this workflow.
name
  string, The name of this workflow.
description
  string, A description of the workflow.
definition
  string, The definition of the workflow in YAML format. Must not be specified if 'fromJobChain' is specified.
valid
  boolean, The validity of the workflow definition.
validationErrors
  string, The errors encountered when validating the workflow definition.
fileId
  string, The file id for the s3 file containing the workflow configuration.
user
  list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user's name.
  • username string, This user's username.
  • initials string, This user's initials.
  • online boolean, Whether this user is online.
state
  string, The state of the workflow. State is "running" if any execution is running, otherwise reflects most recent execution state.
schedule
  list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
• scheduledMinutes array, Minutes of the day it is scheduled on.
• scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.

allowConcurrentExecutions
boolean, Whether the workflow can execute when already running.

timeZone
string, The time zone of this workflow.

nextExecutionAt
string, The time of the next scheduled execution.

notifications
list, A list containing the following elements:
• urls array, URLs to receive a POST request at job completion
• successEmailSubject string, Custom subject line for success e-mail.
• successEmailBody string, Custom body text for success e-mail, written in Markdown.
• successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
• failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on
• failureOn boolean, If failure email notifications are on

archived
string, The archival status of the requested item(s).

hidden
boolean, The hidden status of the item.

createdAt
string,

updatedAt
string,

---

workflows_post_executions

Execute a workflow

Description

Execute a workflow

Usage

workflows_post_executions(
    id,
    target_task = NULL,
    input = NULL,
    included_tasks = NULL
)
Arguments

id integer required. The ID for the workflow.
target_task string optional. For a reverse workflow, the name of the task to target.
input list optional. Key-value pairs to send to this execution as inputs.
included_tasks array optional. If specified, executes only the subset of workflow tasks included.

Value

A list containing the following elements:

id integer, The ID for this workflow execution.
state string, The state of this workflow execution.
mistralState string, The state of this workflow as reported by mistral. One of running, paused, success, error, or cancelled
mistralStateInfo string, The state info of this workflow as reported by mistral.
user list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string. This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
definition string, The definition of the workflow for this execution.
input list, Key-value pairs defined for this execution.
includedTasks array, The subset of workflow tasks selected to execute.
tasks array, An array containing the following fields:
  • name string, The name of the task.
  • mistralState string, The state of this task. One of idle, waiting, running, delayed, success, error, or cancelled
  • mistralStateInfo string, Extra info associated with the state of the task.
  • runs array, The runs associated with this task, in descending order by id.
  • executions array, The executions run by this task, in descending order by id.
startedAt string, The time this execution started.
finishedAt string, The time this execution finished.
createdAt string, The time this execution was created.
updatedAt string, The time this execution was last updated.
workflows_post_executions_cancel

Cancel a workflow execution

Description
Cancel a workflow execution

Usage
workflows_post_executions_cancel(id, execution_id)

Arguments
id integer required. The ID for the workflow.

execution_id integer required. The ID for the workflow execution.

Value
A list containing the following elements:

id integer, The ID for this workflow execution.

state string, The state of this workflow execution.

mistralState string, The state of this workflow as reported by mistral. One of running, paused, success, error, or cancelled

mistralStateInfo string, The state info of this workflow as reported by mistral.

user list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user's name.
  • username string, This user's username.
  • initials string, This user's initials.
  • online boolean, Whether this user is online.

definition string, The definition of the workflow for this execution.

input list, Key-value pairs defined for this execution.

includedTasks array, The subset of workflow tasks selected to execute.

tasks array, An array containing the following fields:
  • name string, The name of the task.
  • mistralState string, The state of this task. One of idle, waiting, running, delayed, success, error, or cancelled
  • mistralStateInfo string, Extra info associated with the state of the task.
  • runs array, The runs associated with this task, in descending order by id.
• executions array, The executions run by this task, in descending order by id.
  startedAt string, The time this execution started.
  finishedAt string, The time this execution finished.
  createdAt string, The time this execution was created.
  updatedAt string, The time this execution was last updated.

---

**workflows_post_executions_resume**

*Resume a paused workflow execution*

**Description**
Resume a paused workflow execution

**Usage**

`workflows_post_executions_resume(id, execution_id)`

**Arguments**

- **id** integer required. The ID for the workflow.
- **execution_id** integer required. The ID for the workflow execution.

**Value**
A list containing the following elements:

- **id** integer, The ID for this workflow execution.
- **state** string, The state of this workflow execution.
- **mistralState** string, The state of this workflow as reported by mistral. One of running, paused, success, error, or cancelled
- **mistralStateInfo** string, The state info of this workflow as reported by mistral.
- **user** list, A list containing the following elements:
  - **id** integer, The ID of this user.
  - **name** string, This user’s name.
  - **username** string, This user’s username.
  - **initials** string, This user’s initials.
  - **online** boolean, Whether this user is online.
- **definition** string, The definition of the workflow for this execution.
- **input** list, Key-value pairs defined for this execution.
- **includedTasks** array, The subset of workflow tasks selected to execute.
workflows_post_executions_retry

Retry a failed task, or all failed tasks in an execution

Description

Retry a failed task, or all failed tasks in an execution

Usage

workflows_post_executions_retry(id, execution_id, task_name = NULL)

Arguments

id integer required. The ID for the workflow.
execution_id integer required. The ID for the workflow execution.
task_name string optional. If specified, the name of the task to be retried. If not specified, all failed tasks in the execution will be retried.

Value

A list containing the following elements:

id integer, The ID for this workflow execution.
state string, The state of this workflow execution.
mistralState string, The state of this workflow as reported by mistral. One of running, paused, success, error, or cancelled
mistralStateInfo string, The state info of this workflow as reported by mistral.
user list, A list containing the following elements:
• id integer, The ID of this user.
• name string, This user’s name.
• username string, This user’s username.
• initials string, This user’s initials.
• online boolean, Whether this user is online.

**definition**
string, The definition of the workflow for this execution.

**input**
list, Key-value pairs defined for this execution.

**includedTasks**
array, The subset of workflow tasks selected to execute.

**tasks**
array, An array containing the following fields:
• name string, The name of the task.
• mistralState string, The state of this task. One of idle, waiting, running, delayed, success, error, or cancelled
• mistralStateInfo string, Extra info associated with the state of the task.
• runs array, The runs associated with this task, in descending order by id.
• executions array, The executions run by this task, in descending order by id.

**startedAt**
string, The time this execution started.

**finishedAt**
string, The time this execution finished.

**createdAt**
string, The time this execution was created.

**updatedAt**
string, The time this execution was last updated.

---

workflows_post_git_commits

Commit and push a new version of the file

**Description**
Commit and push a new version of the file

**Usage**
workflows_post_git_commits(id, content, message, file_hash)

**Arguments**
- id integer required. The ID of the file.
- content string required. The contents to commit to the file.
- message string required. A commit message describing the changes being made.
- file_hash string required. The full SHA of the file being replaced.
workflows_put

Value
A list containing the following elements:

- content string, The file’s contents.
- type string, The file’s type.
- size integer, The file’s size.
- fileHash string, The SHA of the file.

workflows_put Replace all attributes of this Workflow

Description
Replace all attributes of this Workflow

Usage

workflows_put(
  id,
  name,
  description = NULL,
  definition = NULL,
  schedule = NULL,
  allow_concurrent_executions = NULL,
  timeZone = NULL,
  notifications = NULL
)

Arguments

id integer required. The ID for this workflow.
name string required. The name of this workflow.
description string optional. A description of the workflow.
definition string optional. The definition of the workflow in YAML format. Must not be specified if ‘fromJobChain’ is specified.
schedule list optional. A list containing the following elements:
- scheduled boolean. If the item is scheduled.
- scheduledDays array, Day based on numeric value starting at 0 for Sunday.
- scheduledHours array, Hours of the day it is scheduled on.
- scheduledMinutes array, Minutes of the day it is scheduled on.
- scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.
allow_concurrent_executions boolean optional. Whether the workflow can execute when already running.
 workflows_put

time_zone string optional. The time zone of this workflow.
notifications list optional. A list containing the following elements:
  - urls array, URLs to receive a POST request at job completion
  - successEmailSubject string, Custom subject line for success e-mail.
  - successEmailBody string, Custom body text for success e-mail, written in Markdown.
  - successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  - failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  - stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  - successOn boolean, If success email notifications are on
  - failureOn boolean, If failure email notifications are on

Value

A list containing the following elements:

id integer, The ID for this workflow.
name string, The name of this workflow.
description string, A description of the workflow.
definition string, The definition of the workflow in YAML format. Must not be specified if ‘fromJobChain’ is specified.
valid boolean, The validity of the workflow definition.
validationErrors string, The errors encountered when validating the workflow definition.
fileId string, The file id for the s3 file containing the workflow configuration.
user list, A list containing the following elements:
  - id integer, The ID of this user.
  - name string, This user’s name.
  - username string, This user’s username.
  - initials string, This user’s initials.
  - online boolean, Whether this user is online.
state string, The state of the workflow. State is "running" if any execution is running, otherwise reflects most recent execution state.
schedule list, A list containing the following elements:
  - scheduled boolean, If the item is scheduled.
  - scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  - scheduledHours array, Hours of the day it is scheduled on.
  - scheduledMinutes array, Minutes of the day it is scheduled on.
  - scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.
allowConcurrentExecutions
  boolean, Whether the workflow can execute when already running.

timeZone
  string, The time zone of this workflow.

nextExecutionAt
  string, The time of the next scheduled execution.

notifications
  list, A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
  • stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
  • successOn boolean, If success email notifications are on
  • failureOn boolean, If failure email notifications are on

archived
  string, The archival status of the requested item(s).

hidden
  boolean, The hidden status of the item.

createdAt
  string,

updatedAt
  string,

---

workflows_put_archive  Update the archive status of this object

Description

Update the archive status of this object

Usage

workflows_put_archive(id, status)

Arguments

id  integer required. The ID of the object.
status  boolean required. The desired archived status of the object.
Value

A list containing the following elements:

id integer, The ID for this workflow.
name string, The name of this workflow.
description string, A description of the workflow.
definition string, The definition of the workflow in YAML format. Must not be specified if ‘fromJobChain’ is specified.
valid boolean, The validity of the workflow definition.
validationErrors string, The errors encountered when validating the workflow definition.
fileId string, The file id for the s3 file containing the workflow configuration.
user list, A list containing the following elements:
  • id integer, The ID of this user.
  • name string, This user’s name.
  • username string, This user’s username.
  • initials string, This user’s initials.
  • online boolean, Whether this user is online.
state string, The state of the workflow. State is “running” if any execution is running, otherwise reflects most recent execution state.
schedule list, A list containing the following elements:
  • scheduled boolean, If the item is scheduled.
  • scheduledDays array, Day based on numeric value starting at 0 for Sunday.
  • scheduledHours array, Hours of the day it is scheduled on.
  • scheduledMinutes array, Minutes of the day it is scheduled on.
  • scheduledRunsPerHour integer, Alternative to scheduled minutes, number of times to run per hour.
allowConcurrentExecutions boolean, Whether the workflow can execute when already running.
timeZone string, The time zone of this workflow.
nextExecutionAt string, The time of the next scheduled execution.
notifications list, A list containing the following elements:
  • urls array, URLs to receive a POST request at job completion
  • successEmailSubject string, Custom subject line for success e-mail.
  • successEmailBody string, Custom body text for success e-mail, written in Markdown.
  • successEmailAddresses array, Addresses to notify by e-mail when the job completes successfully.
  • failureEmailAddresses array, Addresses to notify by e-mail when the job fails.
• stallWarningMinutes integer, Stall warning emails will be sent after this amount of minutes.
• successOn boolean, If success email notifications are on
• failureOn boolean, If failure email notifications are on

archived string, The archival status of the requested item(s).
hidden boolean, The hidden status of the item.
createdAt string,
updatedAt string,

```
workflows_put_git
  Attach an item to a file in a git repo
```

Description

Attach an item to a file in a git repo

Usage

```
workflows_put_git(
  id,
  git_ref = NULL,
  git_branch = NULL,
  git_path = NULL,
  git_repo_url = NULL,
  pull_from_git = NULL
)
```

Arguments

id integer required. The ID of the file.
git_ref string optional. A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
git_branch string optional. The git branch that the file is on.
git_path string optional. The path of the file in the repository.
git_repo_url string optional. The URL of the git repository.
pull_from_git boolean optional. Automatically pull latest commit from git. Only works for scripts.
Value

A list containing the following elements:

- `gitRef` string, A git reference specifying an unambiguous version of the file. Can be a branch name, or the full or shortened SHA of a commit.
- `gitBranch` string, The git branch that the file is on.
- `gitPath` string, The path of the file in the repository.
- `gitRepo` list, A list containing the following elements:
  - `id` integer, The ID for this git repository.
  - `repoUrl` string, The URL for this git repository.
  - `createdAt` string,
  - `updatedAt` string,
- `pullFromGit` boolean, Automatically pull latest commit from git. Only works for scripts.

workflows_put_projects

Add a Workflow to a project

Description

Add a Workflow to a project

Usage

workflows_put_projects(id, project_id)

Arguments

<table>
<thead>
<tr>
<th>Argument</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td>required. The ID of the Workflow.</td>
</tr>
<tr>
<td>project_id</td>
<td>integer</td>
<td>required. The ID of the project.</td>
</tr>
</tbody>
</table>

Value

An empty HTTP response
workflows_put_shares_groups

*Set the permissions groups has on this object*

**Description**

Set the permissions groups has on this object

**Usage**

```python
workflows_put_shares_groups(
    id,
    group_ids,
    permission_level,
    share_email_body = NULL,
    send_shared_email = NULL
)
```

**Arguments**

- **id**: integer required. The ID of the resource that is shared.
- **group_ids**: array required. An array of one or more group IDs.
- **permission_level**: string required. Options are: "read", "write", or "manage".
- **share_email_body**: string optional. Custom body text for e-mail sent on a share.
- **send_shared_email**: boolean optional. Send email to the recipients of a share.

**Value**

A list containing the following elements:

- **readers**: list, A list containing the following elements:
  - users array,
  - groups array,
- **writers**: list, A list containing the following elements:
  - users array,
  - groups array,
- **owners**: list, A list containing the following elements:
  - users array,
  - groups array,
- **totalUserShares**: integer, For owners, the number of total users shared. For writers and readers, the number of visible users shared.
totalGroupShares
integer, For owners, the number of total groups shared. For writers and readers, the number of visible groups shared.

workflows_put_shares_users
Set the permissions users have on this object

Description
Set the permissions users have on this object

Usage
workflows_put_shares_users(
  id,
  user_ids,
  permission_level,
  share_email_body = NULL,
  send_shared_email = NULL
)

Arguments
id integer required. The ID of the resource that is shared.
user_ids array required. An array of one or more user IDs.
permission_level string required. Options are: "read", "write", or "manage".
share_email_body string optional. Custom body text for e-mail sent on a share.
send_shared_email boolean optional. Send email to the recipients of a share.

Value
A list containing the following elements:
readers list, A list containing the following elements:
  • users array,
  • groups array,
writers list, A list containing the following elements:
  • users array,
  • groups array,
owners list, A list containing the following elements:
  • users array,
write_chunks

Split a file into chunks of a given chunk size, returning a list of file names.

Description

Split a file into chunks of a given chunk size, returning a list of file names.

Usage

write_chunks(file, chunk_size)

Arguments

file name of the file
chunk_size size of the chunk in bytes.

write_civis

Upload a local data frame or csv file to the Civis Platform (Redshift)

Description

Uploads a data frame, a csv file, or file on S3 to Redshift based on the first argument.

A default database can be set using `options(civis.default_db = "my_database")`. If there is only one database available, this database will automatically be used as the default.

Usage

write_civis(x, ...)

## S3 method for class 'data.frame'
write_civis(
x,
tablename,
database = NULL,
if_exists = "fail",
distkey = NULL,)
sortkey1 = NULL,
sortkey2 = NULL,
max_errors = NULL,
verbose = FALSE,
hidden = TRUE,
diststyle = NULL,
header = TRUE,
credential_id = NULL,
import_args = NULL,
...
)

## S3 method for class 'character'
write_civis(
  x,
  tablename,
  database = NULL,
  if_exists = "fail",
distkey = NULL,
  sortkey1 = NULL,
sortkey2 = NULL,
  max_errors = NULL,
  verbose = FALSE,
  hidden = TRUE,
diststyle = NULL,
  header = TRUE,
  credential_id = NULL,
  import_args = NULL,
  ...
)

## S3 method for class 'numeric'
write_civis(
  x,
  tablename,
  database = NULL,
  if_exists = "fail",
distkey = NULL,
  sortkey1 = NULL,
sortkey2 = NULL,
  max_errors = NULL,
  verbose = FALSE,
delimiter = ",",
  hidden = TRUE,
diststyle = NULL,
  header = TRUE,
  credential_id = NULL,
  import_args = NULL,
Arguments

- `x`: data frame, file path of a csv, or the id of a csv file on S3 to upload to platform.
- `tablename`: string, Name of table and schema "schema.tablename".
- `database`: string, Name of database where data frame is to be uploaded. If no database is specified, uses options(civis.default_db).
- `if_exists`: string, optional, String indicating action to take if table already exists. Must be either "fail", "drop", "truncate" or "append". Defaults to "fail".
- `distkey`: string, optional, Column name designating the distkey.
- `sortkey1`: string, optional, Column name designating the first sortkey.
- `sortkey2`: string, optional, Column name designating the second (compound) sortkey.
- `max_errors`: int, optional, Maximum number of rows with errors to remove before failing.
- `verbose`: bool, Set to TRUE to print intermediate progress indicators.
- `hidden`: bool, if TRUE (default), this job will not appear in the Civis UI.
- `diststyle`: string optional. The diststyle to use for the table. One of "even", "all", or "key".
- `header`: bool, if TRUE (default) the first row is a header.
- `credential_id`: integer, the id of the credential to be used when performing the database import. If NULL (default), the default credential of the current user will be used.
- `import_args`: list of additional arguments for imports_post_files.
- `delimiter`: string, optional. Which delimiter to use. One of ',', '＼t' or ' | '.

Methods (by class)

- `data.frame`: Upload a data frame to Civis Platform (Redshift).
- `character`: Upload a csv to Civis Platform (Redshift).
- `numeric`: Upload a csv file from the files endpoint to Civis Platform (Redshift)

See Also

- `refresh_table` to update table meta-data.

Other io: `download_civis()`, `query_civis_file()`, `query_civis()`, `read_civis()`, `write_civis_file()`

Examples

```r
## Not run:
df <- read.csv(local_file)

# Create new table, fail if already exists
write_civis(df, "schema.my_table", "my_database")
```
# Create new table, append if already exists
write_civis(df, "schema.my_table", "my_database", if_exists="append")

# Create new table with additional options
write_civis(df, "schema.my_table", "my_database",
            distkey="id",
            sortkey1="added_date",
            credential_id = 1,
            header = FALSE)

# Create new table directly from a saved csv
write_civis("my/file/path.csv", "schema.my_table", "my_database")

# Create new table from a file_id
id <- write_civis_file("my/file/path.csv", name = "path.csv")
write_civis(id, "schema.my_table", "my_database")

## End(Not run)

---

### write_civis_file

*Upload a R object or file to Civis Platform (Files endpoint)*

**Description**

Uploads a data frame, R object or file to the files endpoint on Civis Platform (Amazon S3). It returns the id of the file for use with `read_civis` or `download_civis`.

Data frames are uploaded as CSVs with `write.csv`. R objects are serialized with `saveRDS`. Files are uploaded as-is. Objects or files larger than 50mb are chunked and can be uploaded in parallel if a `plan` has been set. Files larger than 5TB cannot be uploaded.

**Usage**

```r
write_civis_file(x, ...)
```

## Default S3 method:
```r
write_civis_file(x, name = "r-object.rds", expires_at = NULL, ...)
```

## S3 method for class 'data.frame'
```r
write_civis_file(  
x,  
    name = "data.csv",  
    expires_at = NULL,  
    row.names = FALSE,  
    ...  
)
```

## S3 method for class 'character'
```r
write_civis_file(x, name = x, expires_at = NULL, ...)
```
### `write_civis_file`

#### Arguments

- **x**: R object or path of file to upload.
- **...**: arguments passed to `saveRDS` or `write.csv` for data frames.
- **name**: string, Name of the file or object.
- **expires_at**: string, The date and time the object will expire on in the format "YYYY-MM-DD HH:MM:SS". `expires_at = NULL` allows files to be kept indefinitely.
- **row.names**: default FALSE. Either a logical value indicating whether the row names of `x` are to be written along with `x`, or a character vector of row names to be written.

#### Details

Data frames are uploaded as CSVs using `write.csv`, with `row.names = FALSE` by default. Additional arguments to `write.csv` can be passed through `...`.

By default, R objects are serialized using `saveRDS` before uploading the object to the files endpoint. If given a filepath, the file is uploaded as-is.

#### Value

The file id which can be used to later retrieve the file using `read_civis`.

#### Methods (by class)

- `default`: Serialize R object
- `data.frame`: Upload a data frame as a csv
- `character`: Upload any file

#### See Also

Other io: `download_civis()`, `query_civis_file()`, `query_civis()`, `read_civis()`, `write_civis()`

#### Examples

```r
## Not run:
data(iris)
file_id <- write_civis_file(iris)
read_civis(file_id)

file_id <- write_civis_file("path/to/my.csv")
read_civis(file_id)
read_civis(file_id, using = readr::read_csv)

file_id <- write_civis_file(list(a = 1))
read_civis(file_id, using = readRDS)

# Does not expire
file_id <- write_civis_file(iris, expires_at = NULL)

# Expires on a given date and time
```
file_id <- write_civis_file(iris, expires_at = "2030-01-01")
file_id <- write_civis_file(iris, expires_at = "12:00:00")
file_id <- write_civis_file(iris, expires_at = "2030-01-01 12:00:00")

# Upload a large file in parallel.
library(future)
plan(multisession)
file_id <- write_civis_file("my_large_file")

## End(Not run)

---

**write_job_output**  
*Add a file as a run output if called from a container job*

**Description**

Add a file as a run output if called from a container job

**Usage**

`write_job_output(filename)`

**Arguments**

- `filename`  
  string, name of the file to add as a run output `civis::scripts_post_*_runs_outputs`.

**Details**

Only posts if running on Civis Platform.

**Value**

Returns the filename if not running on platform.
Index

*Topic datasets
   CIVIS_ML_CLASSIFIERS, 68
   CIVIS_ML_REGRESSORS, 94

admin_delete_announcements, 18
admin_get_announcements, 18
admin_list_announcements, 19
admin_list_organizations, 20
admin_patch_announcements, 21
admin_patch_themes, 22
admin_post_announcements, 23
admin_post_themes, 23
aliases_delete, 24
aliases_delete_shares_groups, 25
aliases_delete_shares_users, 25
aliases_get, 26
aliases_get_object_type, 26
aliases_list, 27
aliases_list_shares, 28
aliases_patch, 29
aliases_post, 30
aliases_put, 31
aliases_put_shares_groups, 32
aliases_put_shares_users, 33
announcements_list, 34
apps_delete_instances_projects, 35
apps_delete_instances_shares_groups, 35
apps_delete_instances_shares_users, 36
apps_delete_releases_shares_groups, 36
apps_delete_releases_shares_users, 37
apps_get, 37
apps_get_instances, 38
apps_get_releases, 39
apps_list, 40
apps_list_instances, 40
apps_list_instances_projects, 41
apps_list_instances_shares, 42
apps_list_releases, 43
apps_list_releases_shares, 44
apps_patch_instances, 45
apps_patch_releases, 46
apps_post_instances, 47
apps_post_releases, 48
apps_put_features, 48
apps_put_instances_archive, 49
apps_put_instances_projects, 50
apps_put_instances_shares_groups, 51
apps_put_instances_shares_users, 52
apps_put_releases_archive, 53
apps_put_releases_shares_groups, 54
apps_put_releases_shares_users, 55
await, 56
await_all (await), 56
cancel, 58
cancel.CivisFuture (CivisFuture), 59
civis_file, 61, 67
civis_file_manifest, 61, 67
civis_ml, 61, 62, 108, 271, 272
CIVIS_ML_CLASSIFIERS, 68
civis_ml_extra_trees_classifier, 65, 68
civis_ml_extra_trees_regressor, 65, 73
civis_ml_fetch_existing (civis_ml), 62
civis_ml_gradient_boosting_classifier, 65, 77
civis_ml_gradient_boosting_regressor, 65, 81
civis_ml_random_forest_classifier, 65, 86
civis_ml_random_forest_regressor, 65, 90
CIVIS_ML_REGRESSORS, 94
civis_ml_sparse_linear_regressor, 65, 95
civis_ml_sparse_logistic, 65, 98
civis_ml_sparse_ridge_regressor, 65, 102
civis_platform, 106, 549
civis_script, 107, 249, 513, 549, 550
civis_table, 67, 108
civis_to_multifile_csv, 67, 108, 145
CivisFuture, 59, 106, 548
clusters_delete_kubernetes_partitions, 110
clusters_get_kubernetes, 110
clusters_get_kubernetes_instance_configs, 111
clusters_get_kubernetes_partitions, 113
clusters_list_kubernetes, 114
clusters_list_kubernetes_deployment_stats, 116
clusters_list_kubernetes_deployments, 115
clusters_list_kubernetes_instance_configs, 117
clusters_list_kubernetes_instance_configs_user_statistics, 118
clusters_list_kubernetes_partitions, 119
clusters_patch_kubernetes, 120
clusters_patch_kubernetes_partitions, 121
clusters_post_kubernetes, 122
clusters_post_kubernetes_partitions, 123
coef.civis_ml, 124
credentials_delete_shares_groups, 125
credentials_delete_shares_users, 125
credentials_get, 126
credentials_list, 126
credentials_list_shares, 128
credentials_post, 129
credentials_post_authenticate, 130
credentials_post_temporary, 131
credentials_put, 131
credentials_put_shares_groups, 132
credentials_put_shares_users, 133
databases_delete_whitelist_ips, 134
databases_get, 135
databases_get_whitelist_ips, 135
databases_list, 136
databases_list_advanced_settings, 137
databases_list_schemas, 137
databases_list_tables, 138
databases_list_tables_search, 139
databases_list_whitelist_ips, 140
databases_patch_advanced_settings, 141
databases_post_schemas_scan, 142
databases_post_whitelist_ips, 142
databases_put_advanced_settings, 143
default_credential, 144
download_civis, 144, 510–512, 514, 945–947
download_civis, 945–947
download_civis, 945–947
endpoints_list, 146
enhancements_delete_cass_ncoa_projects, 146
enhancements_delete_cass_ncoa_runs, 147
enhancements_delete_cass_ncoa_shares_groups, 147
enhancements_delete_cass_ncoa_shares_users, 148
enhancements_delete_civis_data_match_projects, 148
enhancements_delete_civis_data_match_runs, 149
enhancements_delete_civis_data_match_shares_groups, 149
enhancements_delete_civis_data_match_shares_users, 150
enhancements_delete_geocode_projects, 150
enhancements_delete_geocode_runs, 151
enhancements_delete_geocode_shares_groups, 151
enhancements_delete_geocode_shares_users, 152
enhancements_get_cass_ncoa, 152
enhancements_get_cass_ncoa_runs, 154
enhancements_get_civis_data_match, 155
enhancements_get_civis_data_match_runs, 157
enhancements_get_geocode, 158
enhancements_get_geocode_runs, 160
enhancements_list, 161
enhancements_list_cass_ncoa_projects, 162
enhancements_list_cass_ncoa_runs, 163
enhancements_list_cass_ncoa_runs_logs, 164
enhancements_list_cass_ncoa_runs_outputs, 165
enhancements_list_cass_ncoa_shares, 166
enhancements_list_civis_data_match_projects, 167
enhancements_list_civis_data_match_runs, 168
enhancements_list_civis_data_match_runs_logs, 169
enhancements_list_civis_data_match_runs_outputs, 170
enhancements_list_civis_data_match_shares, 171
enhancements_list_field_mapping, 172
enhancements_list_geocode_projects, 172
enhancements_list_geocode_runs, 173
enhancements_list_geocode_runs_logs, 174
enhancements_list_geocode_runs_outputs, 175
enhancements_list_geocode_shares, 176
enhancements_list_types, 177
enhancements_patch_cass_ncoa, 177
enhancements_patch_civis_data_match, 181
enhancements_patch_geocode, 185
enhancements_post_cass_ncoa, 188
enhancements_post_cass_ncoa_cancel, 192
enhancements_post_cass_ncoa_runs, 192
enhancements_post_civis_data_match, 193
enhancements_post_civis_data_match_cancel, 197
enhancements_post_civis_data_match_clone, 197
enhancements_post_civis_data_match_runs, 200
enhancements_post_geocode, 200
enhancements_post_geocode_cancel, 204
enhancements_post_geocode_runs, 204
enhancements_put_cass_ncoa, 205
enhancements_put_cass_ncoa_archive, 209
enhancements_put_cass_ncoa_projects, 211
enhancements_put_cass_ncoa_shares_groups, 212
enhancements_put_cass_ncoa_shares_users, 213
enhancements_put_civis_data_match, 214
enhancements_put_civis_data_match_archive, 218
enhancements_put_civis_data_match_projects, 220
enhancements_put_civis_data_match_runs, 220
enhancements_put_civis_data_match_runs_logs, 220
enhancements_put_civis_data_match_runs_outputs, 221
enhancements_put_civis_data_match_shares_groups, 221
enhancements_put_civis_data_match_shares_users, 221
enhancements_put_geocode, 222
enhancements_put_geocode_archive, 226
enhancements_put_geocode_projects, 228
enhancements_put_geocode_runs, 228
enhancements_put_geocode_shares_groups, 228
enhancements_put_geocode_shares_users, 229
environment, 59
exports_get_files_csv, 230
exports_list, 232
exports_patch_files_csv, 233
exports_post_files_csv, 235
exports_put_files_csv, 238
exports_put_files_csv_archive, 240
expression, 59
feature_flags_delete_groups, 241
feature_flags_delete_users, 242
feature_flags_get, 242
feature_flags_list, 243
feature_flags_put_groups, 244
feature_flags_put_users, 245
fetch_all, 246, 250
fetch_and_generate_client, 246
fetch_logs, 57, 58, 67, 247
fetch_logs.CivisFuture (CivisFuture), 59
fetch_oos_scores, 67, 248
fetch_output (fetch_output_file_ids), 248
fetch_output_file_ids, 107, 248, 549, 550
fetch_predictions, 67, 249
fetch_until, 246, 250
files_delete_preprocess_csv, 251
files_delete_projects, 251
files_delete_shares_groups, 252
files_delete_shares_users, 252
files_get, 253
files_get_preprocess_csv, 254
files_list_projects, 255
files_list_shares, 256
files_patch, 257
files_patch_preprocess_csv, 258
files_post, 259
files_post multipart, 260
files_post multipart_complete, 261
files_post preprocess_csv, 261
files_put, 263
files_put preprocess_csv, 264
files_put preprocess_csv archive, 265
files_put projects, 266
files_put shares groups, 267
files_put shares users, 268
Future, 60
future(), 106
future(), 60

generate_client, 269
get_database_id, 269
get_error, 57, 58, 270
get_feature_importance, 271
get_job_type_version, 271
get_metric, 67, 272
get_script_fun, 272
get_status, 57, 58, 273
get_table_id, 274, 515, 892
get_template_ids_all_versions, 274
get_train_template_id, 275
git_repos_delete, 275
git_repos_get, 276
git_repos_list, 276
git_repos_post, 277
groups_delete, 278
groups_delete members, 278
groups_delete shares groups, 279
groups_delete shares users, 279
groups_get, 280
groups_list, 281
groups_list shares, 282
groups_patch, 283
groups_post, 285
groups_put, 287
groups_put members, 289
groups_put shares groups, 290
groups_put shares users, 291

hist.civis_ml, 292
imports_delete files csv, 293
imports_delete files runs, 293

imports_delete projects, 294
imports_delete shares groups, 294
imports_delete shares users, 295
imports_delete syncs, 295
imports_get, 296
imports_get batches, 298
imports_get files csv, 299
imports_get files runs, 300
imports_list, 301
imports_list batches, 304
imports_list files runs, 305
imports_list files runs logs, 306
imports_list projects, 306
imports_list runs, 307
imports_list runs logs, 308
imports_list shares, 309
imports_patch files csv, 310
imports_post, 313
imports_post batches, 317
imports_post cancel, 318
imports_post files, 319, 945
imports_post files csv, 320
imports_post files runs, 324
imports_post runs, 324
imports_post syncs, 325, 892
imports_put, 329
imports_put archive, 333
imports_put files csv, 335
imports_put files csv archive, 339
imports_put projects, 341
imports_put shares groups, 341
imports_put shares users, 342
imports_put syncs, 343
imports_put syncs archive, 348
jobs_delete projects, 350
jobs_delete runs, 351
jobs_delete shares groups, 351
jobs_delete shares users, 352
jobs_get, 352
jobs_get runs, 353
jobs_list, 354
jobs_list children, 356
jobs_list parents, 357
jobs_list projects, 358
jobs_list runs, 359
jobs_list runs logs, 360
jobs_list runs outputs, 361
jobs_list shares, 362
jobs_list_workflows, 363
jobs_post_runs, 364
jobs_post_trigger_email, 365
jobs_put_archive, 365
jobs_put_projects, 367
jobs_put_shares_groups, 367
jobs_put_shares_users, 368
json_values_delete_shares_groups, 369
json_values_delete_shares_users, 370
json_values_get, 370
json_values_list_shares, 371
json_values_patch, 372
json_values_post, 372
json_values_put_shares_groups, 373
json_values_put_shares_users, 374
match_targets_delete_shares_groups, 375
match_targets_delete_shares_users, 376
match_targets_get, 376
match_targets_list, 377
match_targets_list_shares, 377
match_targets_patch, 378
match_targets_post, 379
match_targets_put_archive, 380
match_targets_put_shares_groups, 380
match_targets_put_shares_users, 381
media_delete_optimizations_runs, 382
media_delete_optimizations_shares_groups, 383
media_delete_optimizations_shares_users, 383
media_delete_ratecards_shares_groups, 384
media_delete_ratecards_shares_users, 384
media_delete_spot_orders_shares_groups, 385
media_delete_spot_orders_shares_users, 385
media_get_optimizations, 386
media_get_optimizations_runs, 387
media_get_ratecards, 388
media_get_spot_orders, 389
media_list_dmas, 389
media_list_optimizations, 390
media_list_optimizations_runs, 391
media_list_optimizations_runs_logs, 392
media_list_optimizations_shares, 393
media_list_ratecards, 394
media_list_ratecards_shares, 394
media_list_spot_orders, 395
media_list_spot_orders_shares, 396
media_list_targets, 397
media_patch_optimizations, 397
media_patch_ratecards, 400
media_post_optimizations, 401
media_post_optimizations_clone, 403
media_post_optimizations_runs, 405
media_post_ratecards, 405
media_post_spot_orders, 406
media_put_optimizations_archive, 407
media_put_optimizations_shares_groups, 408
media_put_optimizations_shares_users, 409
media_put_ratecards, 410
media_put_ratecards_archive, 411
media_put_ratecards_shares_groups, 412
media_put_ratecards_shares_users, 413
media_put_spot_orders, 414
media_put_spot_orders_archive, 415
media_put_spot_orders_shares_groups, 415
media_put_spot_orders_shares_users, 416
models_delete_builds, 417
models_delete_projects, 418
models_delete_shares_groups, 418
models_delete_shares_users, 419
models_get, 419
models_get_builds, 422
models_list, 423
models_list_builds, 426
models_list_builds_logs, 428
models_list_projects, 428
models_list_schedules, 429
models_list_shares, 430
models_list_types, 431
models_put_archive, 431
models_put_projects, 434
models_put_shares_groups, 435
models_put_shares_users, 436
multipart_upload, 437
notebooks_delete, 437
notebooks_delete_deployments, 438
notebooks_delete_projects, 438
notebooks_delete_shares_groups, 439
notebooks_delete_shares_users, 439
notebooks_get, 440
notebooks_get_deployments, 442
notebooks_get_git_commits, 443
notebooks_list, 443
notebooks_list_deployments, 445
notebooks_list_deployments_logs, 446
notebooks_list_git, 447
notebooks_list_git_commits, 448
notebooks_list_projects, 448
notebooks_list_shares, 449
notebooks_list_update_links, 450
notebooks_patch, 451
notebooks_post, 453
notebooks_post_clone, 456
notebooks_post_deployments, 458
notebooks_post_git_commits, 459
notebooks_put, 460
notebooks_put_archive, 463
notebooks_put_git, 465
notebooks_put_projects, 466
notebooks_put_shares_groups, 466
notebooks_put_shares_users, 467
notifications_list, 468
ontologies_list, 469
queries_delete_runs, 501
queries_get, 501
queries_get_runs, 502
queries_list, 503
queries_list_runs, 504
queries_list_runs_logs, 505
queries_post, 506
queries_post_runs, 508
queries_put_scripts, 509
query_civis, 145, 510–512, 513, 945–947
query_civis_file, 145, 510, 511, 514, 945, 947
read_civis, 107, 145, 510–512, 513, 945–947
refresh_table, 274, 515, 892, 945
remote_hosts_list, 515
remote_hosts_list_data_sets, 516
remote_hosts_post, 517
remote_hosts_post_authenticate, 518
reports_delete_grants, 518
reports_delete_projects, 519
reports_delete_services_projects, 519
reports_delete_services_shares_groups, 520
reports_delete_services_shares_users, 520
reports_delete_shares_groups, 521
reports_delete_shares_users, 521
reports_get, 522
reports_get_git_commits, 523
reports_get_services, 524
reports_list, 525
reports_list_data_sets, 527
reports_list_git_commits, 527
reports_list_projects, 528
reports_list_services_projects, 529
reports_list_services_shares, 530
reports_list_shares, 531
reports_patch, 532
reports_patch_services, 534
reports_post, 535
reports_post_git_commits, 537
templates_list_reports, 872
templates_list_reports_shares, 873
templates_list_scripts, 874
templates_list_scripts_projects, 875
templates_list_scripts_shares, 876
templates_patch_reports, 877
templates_patch_scripts, 878
templates_post_reports, 879
templates_post_reports_review, 881
templates_post_scripts, 882
templates_post_scripts_review, 883
templates_put_reports, 884
templates_put_reports_shares_groups, 885
templates_put_reports_shares_users, 886
templates_put_scripts, 887
templates_put_scripts_projects, 888
templates_put_scripts_shares_groups, 889
templates_put_scripts_shares_users, 890
transfer_table, 274, 515, 891
users_delete_api_keys, 892
users_delete_me_favorites, 893
users_get, 894
users_get_api_keys, 895
users_list, 896
users_list_api_keys, 897
users_list_me, 898
users_list_me_favorites, 899
users_list_me_ui, 900
users_patch, 901
users_patch_me, 903
users_post, 907
users_post_api_keys, 909
users_post_me_favorites, 910
workflows_delete_projects, 911
workflows_delete_shares_groups, 911
workflows_delete_shares_users, 912
workflows_get, 912
workflows_get_executions, 914
workflows_get_executions_tasks, 915
workflows_get_git_commits, 916
workflows_list, 917
workflows_list_executions, 918
workflows_list_git, 919
workflows_list_git_commits, 920
workflows_list_projects, 921
workflows_list_shares, 922
workflows_patch, 923
workflows_post, 925
workflows_post_clone, 928
workflows_post_executions, 929
workflows_post_executions_cancel, 931
workflows_post_executions_resume, 932
workflows_post_executions_retry, 933
workflows_post_git_commits, 934
workflows_put, 935
workflows_put_archive, 937
workflows_put_git, 939
workflows_put_projects, 940
workflows_put_shares_groups, 941
workflows_put_shares_users, 942
write.csv, 946, 947
write_chunks, 943
write_civis, 145, 510, 512, 514, 943, 947
write_civis_file, 145, 510, 512, 514, 945, 946
write_job_output, 948