Package ‘paws.analytics’

January 12, 2020

Title   Amazon Web Services Analytics APIs
Version 0.1.6
Description Interface to Amazon Web Services analytics APIs, including
    'Elastic MapReduce' 'Hadoop' and 'Spark' big data service,
License Apache License (>= 2.0)
Imports paws.common (>= 0.2.5)
Suggests testthat
Encoding UTF-8
LazyData true
RoxygenNote 7.0.2

Collate 'athena_service.R' 'athena_interfaces.R' 'athena_operations.R'
    'cloudsearch_service.R' 'cloudsearch_interfaces.R'
    'cloudsearch_operations.R' 'cloudsearchdomain_service.R'
    'cloudsearchdomain_interfaces.R'
    'cloudsearchdomain_operations.R' 'datapipeline_service.R'
    'datapipeline_interfaces.R' 'datapipeline_operations.R'
    'elasticsearchservice_service.R'
    'elasticsearchservice_interfaces.R'
    'elasticsearchservice_operations.R' 'emr_service.R'
    'emr_interfaces.R' 'emr_operations.R' 'firehose_service.R'
    'firehose_interfaces.R' 'firehose_operations.R'
    'glue_service.R' 'glue_interfaces.R' 'glue_operations.R'
    'kafka_service.R' 'kafka_interfaces.R' 'kafka_operations.R'
    'kinesis_service.R' 'kinesis_interfaces.R'
    'kinesis_operations.R' 'kinesisanalytics_service.R'
    'kinesisanalytics_interfaces.R' 'kinesisanalytics_operations.R'
    'kinesisanalyticsv2_service.R'
    'kinesisanalyticsv2_interfaces.R'
    'kinesisanalyticsv2_operations.R' 'mturk_service.R'
    'mturk_interfaces.R' 'mturk_operations.R'
    'quicksight_service.R' 'quicksight_interfaces.R'
    'quicksight_operations.R'
Description

Amazon Athena is an interactive query service that lets you use standard SQL to analyze data directly in Amazon S3. You can point Athena at your data in Amazon S3 and run ad-hoc queries and get results in seconds. Athena is serverless, so there is no infrastructure to set up or manage. You pay only for the queries you run. Athena scales automatically—executing queries in parallel—so results are fast, even with large datasets and complex queries. For more information, see What is Amazon Athena in the Amazon Athena User Guide.

If you connect to Athena using the JDBC driver, use version 1.1.0 of the driver or later with the Amazon Athena API. Earlier version drivers do not support the API. For more information and to download the driver, see Accessing Amazon Athena with JDBC.

For code samples using the AWS SDK for Java, see Examples and Code Samples in the Amazon Athena User Guide.
Usage

```
athena(config = list())
```

Arguments

- `config`: Optional configuration of credentials, endpoint, and/or region.

Service syntax

```
svc <- athena(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)
```

Operations

- `batch_get_named_query`: Returns the details of a single named query or a list of up to 50 queries, which you provide as an array of query ID strings.
- `batch_get_query_execution`: Returns the details of a single query execution or a list of up to 50 query executions, which you provide as an array of query execution ID strings.
- `create_named_query`: Creates a named query in the specified workgroup.
- `create_work_group`: Creates a workgroup with the specified name.
- `delete_named_query`: Deletes the named query if you have access to the workgroup in which the query was saved.
- `delete_work_group`: Deletes the workgroup with the specified name.
- `get_named_query`: Returns information about a single query.
- `get_query_execution`: Returns information about a single execution of a query if you have access to the workgroup in which the query ran.
- `get_query_results`: Streams the results of a single query execution specified by QueryExecutionId from the Athena query results location in Amazon S3.
- `get_work_group`: Returns information about the workgroup with the specified name.
- `list_named_queries`: Provides a list of available query IDs only for queries saved in the specified workgroup.
- `list_query_executions`: Provides a list of available query execution IDs for the queries in the specified workgroup.
- `list_tags_for_resource`: Lists the tags associated with this workgroup.
- `list_work_groups`: Lists available workgroups for the account.
- `start_query_execution`: Runs the SQL query statements contained in the Query.
- `stop_query_execution`: Stops a query execution.
- `tag_resource`: Adds one or more tags to the resource, such as a workgroup.
- `untag_resource`: Removes one or more tags from the workgroup resource.
- `update_work_group`: Updates the workgroup with the specified name.
Amazon CloudSearch Configuration Service

You use the Amazon CloudSearch configuration service to create, configure, and manage search domains. Configuration service requests are submitted using the AWS Query protocol. AWS Query requests are HTTP or HTTPS requests submitted via HTTP GET or POST with a query parameter named Action.

The endpoint for configuration service requests is region-specific: `cloudsearch.region.amazonaws.com`. For example, `cloudsearch.us-east-1.amazonaws.com`. For a current list of supported regions and endpoints, see Regions and Endpoints.

Usage

```r
cloudsearch(config = list())
```

Arguments

- `config`: Optional configuration of credentials, endpoint, and/or region.

Service syntax

```r
svc <- cloudsearch(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)
```

Operations

Examples

```r
svc <- athena()
svc$batch_get_named_query(
  Foo = 123
)
```
### Examples

```r
svc <- cloudsearch()
svc$build_suggesters(
  Foo = 123
)
```

---

### Description

You use the AmazonCloudSearch2013 API to upload documents to a search domain and search those documents.

The endpoints for submitting `UploadDocuments`, `Search`, and `Suggest` requests are domain-specific.

To get the endpoints for your domain, use the Amazon CloudSearch configuration service `DescribeDomains`.

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>build_suggesters</code></td>
<td>Indexes the search suggestions</td>
</tr>
<tr>
<td><code>create_domain</code></td>
<td>Creates a new search domain</td>
</tr>
<tr>
<td><code>define_analysis_scheme</code></td>
<td>Configures an analysis scheme that can be applied to a text or text-array field to define language-specific text processing options</td>
</tr>
<tr>
<td><code>define_expression</code></td>
<td>Configures an Expression for the search domain</td>
</tr>
<tr>
<td><code>define_index_field</code></td>
<td>Configures an IndexField for the search domain</td>
</tr>
<tr>
<td><code>define_suggester</code></td>
<td>Configures a suggester for a domain</td>
</tr>
<tr>
<td><code>delete_analysis_scheme</code></td>
<td>Deletes an analysis scheme</td>
</tr>
<tr>
<td><code>delete_domain</code></td>
<td>Permanently deletes a search domain and all of its data</td>
</tr>
<tr>
<td><code>delete_expression</code></td>
<td>Removes an Expression from the search domain</td>
</tr>
<tr>
<td><code>delete_index_field</code></td>
<td>Removes an IndexField from the search domain</td>
</tr>
<tr>
<td><code>delete_suggester</code></td>
<td>Deletes a suggester</td>
</tr>
<tr>
<td><code>describe_analysis_schemes</code></td>
<td>Gets the analysis schemes configured for a domain</td>
</tr>
<tr>
<td><code>describe_availability_options</code></td>
<td>Gets the availability options configured for a domain</td>
</tr>
<tr>
<td><code>describe_domain_endpoint_options</code></td>
<td>Returns the domain’s endpoint options, specifically whether all requests to the domain must arrive over HTTPS</td>
</tr>
<tr>
<td><code>describe_domains</code></td>
<td>Gets information about the search domains owned by this account</td>
</tr>
<tr>
<td><code>describe_expressions</code></td>
<td>Gets the expressions configured for the search domain</td>
</tr>
<tr>
<td><code>describe_index_fields</code></td>
<td>Gets information about the index fields configured for the search domain</td>
</tr>
<tr>
<td><code>describe_scaling_parameters</code></td>
<td>Gets the scaling parameters configured for a domain</td>
</tr>
<tr>
<td><code>describe_service_access_policies</code></td>
<td>Gets information about the access policies that control access to the domain’s documents</td>
</tr>
<tr>
<td><code>describe_suggesters</code></td>
<td>Gets the suggesters configured for a domain</td>
</tr>
<tr>
<td><code>index_documents</code></td>
<td>Tells the search domain to start indexing its documents using the latest indexing options</td>
</tr>
<tr>
<td><code>list_domain_names</code></td>
<td>Lists all search domains owned by an account</td>
</tr>
<tr>
<td><code>update_availability_options</code></td>
<td>Configures the availability options for a domain</td>
</tr>
<tr>
<td><code>update_domain_endpoint_options</code></td>
<td>Updates the domain’s endpoint options, specifically whether all requests to the domain must arrive over HTTPS</td>
</tr>
<tr>
<td><code>update_scaling_parameters</code></td>
<td>Configures scaling parameters for a domain</td>
</tr>
<tr>
<td><code>update_service_access_policies</code></td>
<td>Configures the access rules that control access to the domain’s document and search endpoints</td>
</tr>
</tbody>
</table>
action. The domain endpoints are also displayed on the domain dashboard in the Amazon CloudSearch console. You submit suggest requests to the search endpoint.
For more information, see the Amazon CloudSearch Developer Guide.

Usage

cloudsearchdomain(config = list())

Arguments

cfg Optional configuration of credentials, endpoint, and/or region.

Service syntax

svc <- cloudsearchdomain(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)

Operations

search Retrieves a list of documents that match the specified search criteria
suggest Retrieves autocomplete suggestions for a partial query string
upload_documents Posts a batch of documents to a search domain for indexing

Examples

svc <- cloudsearchdomain()
svc$search(
  Foo = 123
)
**Description**

AWS Data Pipeline configures and manages a data-driven workflow called a pipeline. AWS Data Pipeline handles the details of scheduling and ensuring that data dependencies are met so that your application can focus on processing the data.

AWS Data Pipeline provides a JAR implementation of a task runner called AWS Data Pipeline Task Runner. AWS Data Pipeline Task Runner provides logic for common data management scenarios, such as performing database queries and running data analysis using Amazon Elastic MapReduce (Amazon EMR). You can use AWS Data Pipeline Task Runner as your task runner, or you can write your own task runner to provide custom data management.

AWS Data Pipeline implements two main sets of functionality. Use the first set to create a pipeline and define data sources, schedules, dependencies, and the transforms to be performed on the data. Use the second set in your task runner application to receive the next task ready for processing. The logic for performing the task, such as querying the data, running data analysis, or converting the data from one format to another, is contained within the task runner. The task runner performs the task assigned to it by the web service, reporting progress to the web service as it does so. When the task is done, the task runner reports the final success or failure of the task to the web service.

**Usage**

```r
datapipeline(config = list())
```

**Arguments**

- `config` Optional configuration of credentials, endpoint, and/or region.

**Service syntax**

```r
csvc <- datapipeline(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)
```
Operations
activate_pipeline Validates the specified pipeline and starts processing pipeline tasks
add_tags Adds or modifies tags for the specified pipeline
create_pipeline Creates a new, empty pipeline
deactivate_pipeline Deactivates the specified running pipeline
delete_pipeline Deletes a pipeline, its pipeline definition, and its run history
describe_objects Gets the object definitions for a set of objects associated with the pipeline
describe_pipelines Retrieves metadata about one or more pipelines
evaluate_expression Task runners call EvaluateExpression to evaluate a string in the context of the specified object
get_pipeline_definition Gets the definition of the specified pipeline
list_pipelines Lists the pipeline identifiers for all active pipelines that you have permission to access
poll_for_task Task runners call PollForTask to receive a task to perform from AWS Data Pipeline
put_pipeline_definition Adds tasks, schedules, and preconditions to the specified pipeline
query_objects Queries the specified pipeline for the names of objects that match the specified set of conditions
remove_tags Removes existing tags from the specified pipeline
report_task_progress Task runners call ReportTaskProgress when assigned a task to acknowledge that it has the task
report_task_runner_heartbeat Task runners call ReportTaskRunnerHeartbeat every 15 minutes to indicate that they are operational
set_status Requests that the status of the specified physical or logical pipeline objects be updated in the specified pipeline
set_task_status Task runners call SetTaskStatus to notify AWS Data Pipeline that a task is completed and provide information about the final status
validate_pipeline_definition Validates the specified pipeline definition to ensure that it is well formed and can be run without error.

Examples

svc <- datapipeline()
svc$activate_pipeline(
  Foo = 123
)

---

elasticsearchservice  Amazon Elasticsearch Service

Description

Amazon Elasticsearch Configuration Service

Use the Amazon Elasticsearch Configuration API to create, configure, and manage Elasticsearch domains.

For sample code that uses the Configuration API, see the Amazon Elasticsearch Service Developer Guide. The guide also contains sample code for sending signed HTTP requests to the Elasticsearch APIs.

The endpoint for configuration service requests is region-specific: es.region.amazonaws.com. For example, es.us-east-1.amazonaws.com. For a current list of supported regions and endpoints, see Regions and Endpoints.
Usage

elasticsearchservice(config = list())

Arguments

cfg Optional configuration of credentials, endpoint, and/or region.

Service syntax

svc <- elasticsearchservice(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)

Operations

add_tags Attaches tags to an existing Elasticsearch domain
cancel_elasticsearch_service_software_update Cancels a scheduled service software update for an Amazon ES domain
create_elasticsearch_domain Creates a new Elasticsearch domain
delete_elasticsearch_domain Permanently deletes the specified Elasticsearch domain and all of its data
delete_elasticsearch_service_role Deletes the service-linked role that Elasticsearch Service uses to manage and maintain VPC domains
describe_elasticsearch_domain Returns domain configuration information about the specified Elasticsearch domain, including the domain ID, domain endpoint, and domain ARN
describe_elasticsearch_domain_config Provides cluster configuration information about the specified Elasticsearch domain, such as the state, creation date, update version, and update date for cluster options
describe_elasticsearch_domains Returns domain configuration information about the specified Elasticsearch domains, including the domain ID, domain endpoint, and domain ARN
describe_elasticsearch_instance_type_limits Describe Elasticsearch Limits for a given InstanceType and ElasticsearchVersion
describe_reserved_elasticsearch_instance_offerings Lists available reserved Elasticsearch instance offerings
describe_reserved_elasticsearch_instances Returns information about reserved Elasticsearch instances for this account
describe_compatible_elasticsearch_versions Returns a list of upgrade compatible Elasticsearch versions
describe_upgrade_history Retrieves the complete history of the last 10 upgrades that were performed
describe_upgrade_status Retrieves the latest status of the last upgrade or upgrade eligibility check
describe_list_domain_names Returns the name of all Elasticsearch domains owned by the current user
describe_list_elasticsearch_instance_types List all Elasticsearch instance types that are supported for given ElasticsearchVersion
describe_list_elasticsearch_versions List all supported Elasticsearch versions
describe_list_tags Returns all tags for the given Elasticsearch domain
describe_purchase_elasticsearch_instance_offerings Allows you to purchase reserved Elasticsearch instances
describe_remove_tags Removes the specified set of tags from the specified Elasticsearch domain
describe_update_elasticsearch_domain_config Schedules a service software update for an Amazon ES domain
describe_update_elasticsearch_domain_modifies The cluster configuration of the specified Elasticsearch domain
upgrade_elasticsearch_domain

Allows you to either upgrade your domain or perform an Upgrade eligibility check to a compatible Elasticsearch version.

Examples

```r
svc <- elasticsearchservice()
svc$add_tags(
    Foo = 123
)
```

Amazon Elastic MapReduce

Description

Amazon EMR is a web service that makes it easy to process large amounts of data efficiently. Amazon EMR uses Hadoop processing combined with several AWS products to do tasks such as web indexing, data mining, log file analysis, machine learning, scientific simulation, and data warehousing.

Usage

```r
emr(config = list())
```

Arguments

- **config**: Optional configuration of credentials, endpoint, and/or region.

Service syntax

```r
svc <- emr(
    config = list(
        credentials = list(
            creds = list(
                access_key_id = "string",
                secret_access_key = "string",
                session_token = "string"
            ),
            profile = "string"
        ),
        endpoint = "string",
        region = "string"
    )
)
```

Operations
add_instance_fleet
add_instance_groups
add_job_flow_steps
add_tags
cancel_steps
create_security_configuration
delete_security_configuration
describe_cluster
describe_job_flows
describe_security_configuration
describe_step
get_block_public_access_configuration
list_bootstrap_actions
list_clusters
list_instance_fleets
list_instance_groups
list_instances
list_security_configurations
list_steps
modify_cluster
modify_instance_fleet
modify_instance_groups
put_auto_scaling_policy
put_block_public_access_configuration
remove_auto_scaling_policy
remove_tags
run_job_flow
set_termination_protection
set_visible_to_all_users
terminate_job_flows

Adds an instance fleet to a running cluster
Adds one or more instance groups to a running cluster
AddJobFlowSteps adds new steps to a running cluster
Adds tags to an Amazon EMR resource
Cancels a pending step or steps in a running cluster
Creates a security configuration, which is stored in the service and can be specified when a cluster is created.
Deletes a security configuration
Provides cluster-level details including status, hardware and software configuration, VPC settings, and so on.
This API is deprecated and will eventually be removed
Provides the details of a security configuration by returning the configuration JSON
Provides more detail about the cluster step
Returns the Amazon EMR block public access configuration for your AWS account
Provides information about the bootstrap actions associated with a cluster
Provides the status of all clusters visible to this AWS account
Lists all available details about the instance fleets in a cluster
Provides all available details about the instance groups in a cluster
Provides information for all active EC2 instances and EC2 instances terminated in the last 30 days
Lists all the security configurations visible to this account, providing their creation dates and times, and their names.
Provides a list of steps for the cluster in reverse order unless you specify stepIds within the request
Modifies the number of steps that can be executed concurrently for the cluster specified using ClusterID
Modifies the target On-Demand and target Spot capacities for the instance fleet with the specified InstanceFleetID within the cluster specified using ClusterID
Creates or updates an automatic scaling policy for a core instance group or task instance group in an Amazon EMR cluster
Creates or updates an Amazon EMR block public access configuration for your AWS account
Removes an automatic scaling policy from a specified instance group within an EMR cluster
Removes tags from an Amazon EMR resource
RunJobFlow creates and starts running a new cluster (job flow)
SetTerminationProtection locks a cluster (job flow) so the EC2 instances in the cluster cannot be terminated by user intervention, an API call, or in the event of a job-flow error
Sets the Cluster$VisibleToAllUsers value, which determines whether the cluster is visible to all IAM users for the AWS account associated with the cluster
TerminateJobFlows shuts a list of clusters (job flows) down

Examples

svc <- emr()
svc$add_instance_fleet(
   Foo = 123
)

Amazon Kinesis Firehose
Description

Amazon Kinesis Data Firehose API Reference

Amazon Kinesis Data Firehose is a fully managed service that delivers real-time streaming data to destinations such as Amazon Simple Storage Service (Amazon S3), Amazon Elasticsearch Service (Amazon ES), Amazon Redshift, and Splunk.

Usage

firehose(config = list())

Arguments

config Optional configuration of credentials, endpoint, and/or region.

Service syntax

svc <- firehose(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)

Operations

create_delivery_stream Creates a Kinesis Data Firehose delivery stream
delete_delivery_stream Deletes a delivery stream and its data
describe_delivery_stream Describes the specified delivery stream and its status
list_delivery_streams Lists your delivery streams in alphabetical order of their names
list_tags_for_delivery_stream Lists the tags for the specified delivery stream
put_record Writes a single data record into an Amazon Kinesis Data Firehose delivery stream
put_record_batch Writes multiple data records into a delivery stream in a single call, which can achieve high
start_delivery_stream_encryption Enables server-side encryption (SSE) for the delivery stream
stop_delivery_stream_encryption Disables server-side encryption (SSE) for the delivery stream
tag_delivery_stream Adds or updates tags for the specified delivery stream
untag_delivery_stream Removes tags from the specified delivery stream
update_destination Updates the specified destination of the specified delivery stream
Examples

```r
svc <- firehose()
svc$create_delivery_stream(
   Foo = 123
)
```

---

**glue**  
*AWS Glue*

**Description**

Defines the public endpoint for the AWS Glue service.

**Usage**

```r
glue(config = list())
```

**Arguments**

- `config`  
  Optional configuration of credentials, endpoint, and/or region.

**Service syntax**

```r
svc <- glue(
   config = list(
      credentials = list(
         creds = list(
            access_key_id = "string",
            secret_access_key = "string",
            session_token = "string"
         ),
         profile = "string"
      ),
      endpoint = "string",
      region = "string"
   )
)
```

**Operations**

- `batch_create_partition`  
  Creates one or more partitions in a batch operation
- `batch_delete_connection`  
  Deletes a list of connection definitions from the Data Catalog
- `batch_delete_partition`  
  Deletes one or more partitions in a batch operation
- `batch_delete_table`  
  Deletes multiple tables at once
- `batch_delete_table_version`  
  Deletes a specified batch of versions of a table
- `batch_get_crawlers`  
  Returns a list of resource metadata for a given list of crawler names
batch_get_dev_endpoints
batch_get_jobs
batch_get_partition
batch_get_triggers
batch_get_workflows
batch_stop_job_run
cancel_ml_task_run
create_classifier
create_connection
create_crawler
create_database
create_dev_endpoint
create_job
create_ml_transform
create_partition
create_script
create_security_configuration
create_table
create_trigger
create_user_defined_function
create_workflow
delete_classifier
delete_connection
delete_crawler
delete_database
delete_dev_endpoint
delete_job
delete_ml_transform
delete_partition
delete_resource_policy
delete_security_configuration
delete_table
delete_table_version
delete_trigger
delete_user_defined_function
delete_workflow
get_catalog_import_status
get_classifier
get_classifiers
get_connection
get_connections
get_crawler
get_crawler_metrics
get_crawlers
get_database
get_databases
get_data_catalog_encryption_settings
get_dataflow_graph

Returns a list of resource metadata for a given list of development endpoint names.
Returns a list of resource metadata for a given list of job names.
Retrieves partitions in a batch request.
Returns a list of resource metadata for a given list of trigger names.
Returns a list of resource metadata for a given list of workflow names.
Stops one or more job runs for a specified job definition.
Cancels (stops) a task run.
Creates a classifier in the user’s account.
Creates a connection definition in the Data Catalog.
Creates a new crawler with specified targets, role, configuration, and optional schedule.
Creates a new database in a Data Catalog.
Creates a new development endpoint.
Creates a new job definition.
Creates an AWS Glue machine learning transform.
Creates a new partition.
Transforms a directed acyclic graph (DAG) into code.
Creates a new security configuration.
Creates a new table definition in the Data Catalog.
Creates a new trigger.
Creates a new function definition in the Data Catalog.
Creates a new workflow.
Removes a classifier from the Data Catalog.
Deletes a connection from the Data Catalog.
Removes a specified crawler from the AWS Glue Data Catalog, unless the crawler state is "STOPPED" or "FAILED".
Removes a specified database from a Data Catalog.
Deletes a specified development endpoint.
Deletes a specified job definition.
Deletes an AWS Glue machine learning transform.
Deletes a specified partition.
Deletes a specified policy.
Deletes a specified security configuration.
Removes a table definition from the Data Catalog.
Deletes a specified version of a table.
Deletes a specified trigger.
Deletes an existing function definition from the Data Catalog.
Deletes a workflow.
Retrieves the status of a migration operation.
Retrieve a classifier by name.
Lists all classifier objects in the Data Catalog.
Retrieves a connection definition from the Data Catalog.
Retrieves a list of connection definitions from the Data Catalog.
Retrieves metadata for a specified crawler.
Retrieves metrics about specified crawlers.
Retrieves metadata for all crawlers defined in the customer account.
Retrieves the definition of a specified database.
Retrieves all databases defined in a given Data Catalog.
Retrieves the security configuration for a specified catalog.
Transforms a Python script into a directed acyclic graph (DAG).
get_dev_endpoint
get_dev_endpoints
get_job
get_job_bookmark
get_job_run
get_job_runs
get_jobs
get_mapping
get_ml_task_run
get_ml_task_runs
get_ml_transform
get_ml_transforms
get_partition
get_partitions
get_plan
get_resource_policy
get_security_configuration
get_security_configurations
get_table
get_tables
get_table_version
get_table_versions
get_tags
get_trigger
get_triggers
get_user_defined_function
get_user_defined_functions
get_workflow
get_workflow_run
get_workflow_run_properties
get_workflow_runs
import_catalog_to_glue
list_crawlers
list_dev_endpoints
list_jobs
list_triggers
list_workflows
put_data_catalog_encryption_settings
put_resource_policy
put_workflow_run_properties
reset_job_bookmark
search_tables
start_crawler
start_crawler_schedule
start_export_labels_task_run
start_import_labels_task_run
start_job_run
start_ml_evaluation_task_run

Retrieves information about a specified development endpoint
Retrieves all the development endpoints in this AWS account
Retrieves an existing job definition
Returns information on a job bookmark entry
Retrieves the metadata for a given job run
Retrieves metadata for all runs of a given job definition
Retrieves all current job definitions
 Creates mappings
 Gets details for a specific task run on a machine learning transform
 Gets a list of runs for a machine learning transform
 Gets an AWS Glue machine learning transform artifact and all its corresponding metadata
 Gets a sortable, filterable list of existing AWS Glue machine learning transforms
 Retrieves information about a specified partition
 Retrieves information about the partitions in a table
 Gets code to perform a specified mapping
 Retrieves a specified resource policy
Retrieves a specified security configuration
Retrieves a list of all security configurations
Retrieves the Table definition in a Data Catalog for a specified table
Retrieves the definitions of some or all of the tables in a given Database
Retrieves a specified version of a table
Retrieves a list of strings that identify available versions of a specified table
Retrieves a list of tags associated with a resource
Retrieves the definition of a trigger
 Gets all the triggers associated with a job
Retrieves a specified function definition from the Data Catalog
Retrieves multiple function definitions from the Data Catalog
Retrieves resource metadata for a workflow
Retrieves the metadata for a given workflow run
Retrieves the workflow run properties which were set during the run
Retrieves metadata for all runs of a given workflow
Imports an existing Amazon Athena Data Catalog to AWS Glue
Retrieves the names of all crawler resources in this AWS account, or the resources with the specified tag
Retrieves the names of all DevEndpoint resources in this AWS account, or the resources with the specified tag
Retrieves the names of all job resources in this AWS account, or the resources with the specified tag
Retrieves the names of all trigger resources in this AWS account, or the resources with the specified tag
Lists names of workflows created in the account
Sets the security configuration for a specified catalog
Sets the Data Catalog resource policy for access control
Puts the specified workflow run properties for the given workflow run
Resets a bookmark entry
Searches a set of tables based on properties in the table metadata as well as on the
Starts a crawl using the specified crawler, regardless of what is scheduled
Changes the schedule state of the specified crawler to SCHEDULED, unless the
Begins an asynchronous task to export all labeled data for a particular transform
Enables you to provide additional labels (examples of truth) to be used to teach the
Starts a job run using a job definition
Starts a task to estimate the quality of the transform
<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>start_ml_labeling_set_generation_task_run</code></td>
<td>Starts the active learning workflow for your machine learning transform to improve the transform's quality by generating label sets and adding labels</td>
</tr>
<tr>
<td><code>start_trigger</code></td>
<td>Starts an existing trigger</td>
</tr>
<tr>
<td><code>start_workflow_run</code></td>
<td>Starts a new run of the specified workflow</td>
</tr>
<tr>
<td><code>stop_crawler</code></td>
<td>If the specified crawler is running, stops the crawl</td>
</tr>
<tr>
<td><code>stop_crawler_schedule</code></td>
<td>Sets the schedule state of the specified crawler to NOT_SCHEDULED, but does not stop the crawler if it is already running</td>
</tr>
<tr>
<td><code>stop_trigger</code></td>
<td>Stops a specified trigger</td>
</tr>
<tr>
<td><code>tag_resource</code></td>
<td>Adds tags to a resource</td>
</tr>
<tr>
<td><code>untag_resource</code></td>
<td>Removes tags from a resource</td>
</tr>
<tr>
<td><code>update_classifier</code></td>
<td>Modifies an existing classifier (a GrokClassifier, an XMLClassifier, a JsonClassifier, or a CsvClassifier, depending on which field is present)</td>
</tr>
<tr>
<td><code>update_connection</code></td>
<td>Updates a connection definition in the Data Catalog</td>
</tr>
<tr>
<td><code>update_crawler</code></td>
<td>Updates a crawler</td>
</tr>
<tr>
<td><code>update_crawler_schedule</code></td>
<td>Updates the schedule of a crawler using a cron expression</td>
</tr>
<tr>
<td><code>update_database</code></td>
<td>Updates an existing database definition in a Data Catalog</td>
</tr>
<tr>
<td><code>update_dev_endpoint</code></td>
<td>Updates a specified development endpoint</td>
</tr>
<tr>
<td><code>update_job</code></td>
<td>Updates an existing job definition</td>
</tr>
<tr>
<td><code>update_ml_transform</code></td>
<td>Updates an existing machine learning transform</td>
</tr>
<tr>
<td><code>update_partition</code></td>
<td>Updates a partition</td>
</tr>
<tr>
<td><code>update_table</code></td>
<td>Updates a metadata table in the Data Catalog</td>
</tr>
<tr>
<td><code>update_trigger</code></td>
<td>Updates a trigger definition</td>
</tr>
<tr>
<td><code>update_user_defined_function</code></td>
<td>Updates an existing function definition in the Data Catalog</td>
</tr>
<tr>
<td><code>update_workflow</code></td>
<td>Updates an existing workflow</td>
</tr>
</tbody>
</table>

**Examples**

```r
svc <- glue()
svc$batch_create_partition(
  Foo = 123
)
```

---

### Description

**Managed Streaming for Kafka**

**Usage**

```r
kafka(config = list())
```

**Arguments**

- `config` Optional configuration of credentials, endpoint, and/or region.
Service syntax

```
svc <- kafka(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)
```

Operations

- **create_cluster**
  Creates a new MSK cluster
- **create_configuration**
  Creates a new MSK configuration
- **delete_cluster**
  Deletes the MSK cluster specified by the Amazon Resource Name (ARN) in the request
- **describe_cluster**
  Returns a description of the MSK cluster whose Amazon Resource Name (ARN) is specified in the request
- **describe_cluster_operation**
  Returns a description of the cluster operation specified by the ARN
- **describe_configuration**
  Returns a description of this MSK configuration
- **describe_configuration_revision**
  Returns a description of this revision of the configuration
- **get_bootstrap_brokers**
  A list of brokers that a client application can use to bootstrap
- **list_cluster_operations**
  Returns a list of all the operations that have been performed on the specified MSK cluster
- **list_clusters**
  Returns a list of all the MSK clusters in the current Region
- **list_configuration_revisions**
  Returns a list of all the MSK configurations in this Region
- **list_configurations**
  Returns a list of all the MSK configurations in this Region
- **list_nodes**
  Returns a list of the broker nodes in the cluster
- **list_tags_for_resource**
  Returns a list of the tags associated with the specified resource
- **tag_resource**
  Adds tags to the specified MSK resource
- **untag_resource**
  Removes the tags associated with the keys that are provided in the query
- **update_broker_count**
  Updates the number of broker nodes in the cluster
- **update_broker_storage**
  Updates the EBS storage associated with MSK brokers
- **update_cluster_configuration**
  Updates the cluster with the configuration that is specified in the request body
- **update_monitoring**
  Updates the monitoring settings for the cluster

Examples

```
svc <- kafka()
svc$create_cluster(
  Foo = 123
)
```
Amazon Kinesis

Description

Amazon Kinesis Data Streams Service API Reference

Amazon Kinesis Data Streams is a managed service that scales elastically for real-time processing of streaming big data.

Usage

kinesis(config = list())

Arguments

config Optional configuration of credentials, endpoint, and/or region.

Service syntax

svc <- kinesis(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)

Operations

- add_tags_to_stream Adds or updates tags for the specified Kinesis data stream
- create_stream Creates a Kinesis data stream
- decrease_stream_retention_period Decreases the Kinesis data stream’s retention period, which is the length of time data records are retained in the stream
- delete_stream Deletes a Kinesis data stream and all its shards and data
- deregister_stream_consumer To deregister a consumer, provide its ARN
- describe_limits Describes the shard limits and usage for the account
- describe_stream Describes the specified Kinesis data stream
- describe_stream_consumer To get the description of a registered consumer, provide the ARN of the consumer
- describe_stream_summary Provides a summarized description of the specified Kinesis data stream without the shard list
- disable_enhanced_monitoring Disables enhanced monitoring
- enable_enhanced_monitoring Enables enhanced Kinesis data stream monitoring for shard-level metrics
### Examples

```r
svc <- kinesis()
svc$add_tags_to_stream(
  Foo = 123
)
```

### Description

#### Overview

This documentation is for version 1 of the Amazon Kinesis Data Analytics API, which only supports SQL applications. Version 2 of the API supports SQL and Java applications. For more information about version 2, see Amazon Kinesis Data Analytics API V2 Documentation.

This is the Amazon Kinesis Analytics v1 API Reference. The Amazon Kinesis Analytics Developer Guide provides additional information.

### Usage

```r
kinesisanalytics(config = list())
```

### Arguments

- **config** (Optional) configuration of credentials, endpoint, and/or region.
Service syntax

```r
svc <- kinesisanalytics(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)
```

Operations

- `add_application_cloud_watch_logging_option`
- `add_application_input`
- `add_application_input_processing_configuration`
- `add_application_output`
- `add_application_reference_data_source`
- `create_application`
- `delete_application`
- `delete_application_cloud_watch_logging_option`
- `delete_application_input_processing_configuration`
- `delete_application_output`
- `delete_application_reference_data_source`
- `describe_application`
- `discover_input_schema`
- `list_applications`
- `list_tags_for_resource`
- `start_application`
- `stop_application`
- `tag_resource`
- `untag_resource`
- `update_application`

Examples

```r
svc <- kinesisanalytics()
svc$add_application_cloud_watch_logging_option(
  Foo = 123
)
```
Amazon Kinesis Analytics

Description

Amazon Kinesis Data Analytics is a fully managed service that you can use to process and analyze streaming data using SQL or Java. The service enables you to quickly author and run SQL or Java code against streaming sources to perform time series analytics, feed real-time dashboards, and create real-time metrics.

Usage

kinesisanalyticsv2(config = list())

Arguments

config Optional configuration of credentials, endpoint, and/or region.

Service syntax

svc <- kinesisanalyticsv2(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)

Operations

add_application_cloud_watch_logging_option Adds an Amazon CloudWatch log stream to monitor application configuration errors
add_application_input Adds a streaming source to your SQL-based Amazon Kinesis Data Analytics application
add_application_input_processing_configuration Adds an InputProcessingConfiguration to an SQL-based Kinesis Data Analytics application
add_application_output Adds an external destination to your SQL-based Amazon Kinesis Data Analytics application
add_application_reference_data_source Adds a reference data source to an existing SQL-based Amazon Kinesis Data Analytics application
add_application_vpc_configuration Adds a Virtual Private Cloud (VPC) configuration to the application
create_application Creates an Amazon Kinesis Data Analytics application
create_application_snapshot Creates a snapshot of the application’s state data
delete_application Deletes the specified application
delete_application_cloud_watch_logging_option Deletes an Amazon CloudWatch log stream from an Amazon Kinesis Data Analytics application
delete_application_input_processing_configuration
delete_application_output
delete_application_reference_data_source
delete_application_snapshot
delete_application_vpc_configuration
describe_application
describe_application_snapshot
discover_input_schema
list_applications
list_application_snapshots
list_tags_for_resource
start_application
stop_application
tag_resource
untag_resource
update_application

Deletes an InputProcessingConfiguration from an input
Deletes the output destination configuration from your SQL-based Amazon Kinesis Data Analytics application
Deletes a reference data source configuration from the specified SQL-based Amazon Kinesis Data Analytics application
Deletes a snapshot of application state
Removes a VPC configuration from a Kinesis Data Analytics application
Returns information about a specific Amazon Kinesis Data Analytics application
Returns information about a snapshot of application state data
Infers a schema for an SQL-based Amazon Kinesis Data Analytics application
Returns a list of Amazon Kinesis Data Analytics applications in your account
Lists information about the current application snapshots
Retrieves the list of key-value tags assigned to the application
Starts the specified Amazon Kinesis Data Analytics application
Stops the application from processing data
Adds one or more key-value tags to a Kinesis Analytics application
Removes one or more tags from a Kinesis Analytics application
Updates an existing Amazon Kinesis Data Analytics application

Examples

```
svc <- kinesisanalyticsv2()
svc$add_application_cloud_watch Logging_option(
  Foo = 123
)
```

Description

Amazon Mechanical Turk API Reference

Usage

```
mturk(config = list())
```

Arguments

config Optional configuration of credentials, endpoint, and/or region.
**Service syntax**

```r
svc <- mturk(
  config = list(
    credentials = list(
      creds = list(
        access_key_id = "string",
        secret_access_key = "string",
        session_token = "string"
      ),
      profile = "string"
    ),
    endpoint = "string",
    region = "string"
  )
)
```

**Operations**

- `accept_qualification_request`: The `accept_qualification_request` operation approves a Worker's request for a Qualification.
- `approve_assignment`: The `approve_assignment` operation approves the results of a completed assignment.
- `associate_qualification_with_worker`: The `associate_qualification_with_worker` operation gives a Worker a Qualification.
- `create_additional_assignments_for_hit`: The `create_additional_assignments_for_hit` operation increases the maximum number of assignments of an existing HIT.
- `create_hit`: The `create_hit` operation creates a new Human Intelligence Task (HIT).
- `create_hit_type`: The `create_hit_type` operation creates a new HIT type.
- `create_hit_with_hit_type`: The `create_hit_with_hit_type` operation creates a new Human Intelligence Task (HIT) using an existing HITTypeID generated by the `create_hit_type` operation.
- `create_qualification_type`: The `create_qualification_type` operation creates a new Qualification type, which is represented by a QualificationType data structure.
- `create_worker_block`: The `create_worker_block` operation allows you to prevent a Worker from working on your HITs.
- `delete_hit`: The `delete_hit` operation is used to delete HIT that is no longer needed.
- `delete_qualification_type`: The `delete_qualification_type` operation deletes a Qualification type and deletes any HIT types that are associated with the Qualification type.
- `delete_worker_block`: The `delete_worker_block` operation allows you to reinstate a blocked Worker to work on your HITs.
- `disassociate_qualification_from_worker`: The `disassociate_qualification_from_worker` revokes a previously granted Qualification from a user.
- `get_account_balance`: The `get_account_balance` operation retrieves the amount of money in your Amazon Mechanical Turk account.
- `get_assignment`: The `get_assignment` operation retrieves the details of the specified Assignment.
- `get_file_upload_url`: The `get_file_upload_url` operation generates and returns a temporary URL.
- `get_hit`: The `get_hit` operation retrieves the details of the specified HIT.
- `get_qualification_score`: The `get_qualification_score` operation returns the value of a Worker's Qualification for a given Qualification type.
- `get_qualification_type`: The `get_qualification_type` operation retrieves information about a Qualification type.
- `list_assignments_for_hit`: The `list_assignments_for_hit` operation retrieves completed assignments for a HIT.
- `list_bonus_payments`: The `list_bonus_payments` operation retrieves the amounts of bonuses you have paid to Workers for a given HIT or assignment.
- `list_hi_ts`: The `list_hi_ts` operation retrieves a list of all HITs.
- `list_hi_ts_for_qualification_type`: The `list_hi_ts_for_qualification_type` operation returns the HITs that use the given Qualification type.
- `list_qualification_requests`: The `list_qualification_requests` operation retrieves requests for Qualifications of a particular Qualification type.
- `list_qualification_types`: The `list_qualification_types` operation returns a list of Qualification types, filtered by the specified search term.
- `list_reviewable_hi_ts`: The `list_reviewable_hi_ts` operation retrieves the HITs with Status equal to Reviewable.
- `list_reviewable_hi_ts_for_qualification_type`: The `list_reviewable_hi_ts_for_qualification_type` operation returns the HITs that use the given Qualification type and have Status equal to Reviewable.
- `list_review_policy_results_for_hit`: The `list_review_policy_results_for_hit` operation retrieves the computed results and the actions taken in the course of executing your Review Policies for a given HIT.
- `list_workers`: The `list_workers` operation retrieves a list of Workers.
- `list_workers_blocks`: The `list_workers_blocks` operation retrieves a list of Workers who are blocked from working on your HITs.
- `list_workers_with_qualification_type`: The `list_workers_with_qualification_type` operation returns all of the Workers that have been associated with a given Qualification type.
- `notify_workers`: The `notify_workers` operation sends an email to one or more Workers that you specify.
**Examples**

```r
svc <- mturk()
svc$accept_qualification_request(
  Foo = 123
)
```

---

**Description**

Amazon QuickSight API Reference

Amazon QuickSight is a fully managed, serverless business intelligence service for the AWS Cloud that makes it easy to extend data and insights to every user in your organization. This API reference contains documentation for a programming interface that you can use to manage Amazon QuickSight.

**Usage**

```r
quicksight(config = list())
```

**Arguments**

- `config`: Optional configuration of credentials, endpoint, and/or region.

**Service syntax**

```r
csvc <- quicksight(
c  config = list(
c    credentials = list(
c      creds = list(
c        access_key_id = "string",
c        secret_access_key = "string",
```
```python
    session_token = "string"
    ),
    profile = "string"
    ),
    endpoint = "string",
    region = "string"
    )
  )

Operations

cancel_ingestion
create_dashboard
create_data_set
create_data_source
create_group
create_group_membership
create_iam_policy_assignment
create_ingestion
create_template
create_template_alias
delete_dashboard
delete_data_set
delete_data_source
delete_group
delete_group_membership
delete_iam_policy_assignment
delete_template
delete_template_alias
delete_user
delete_user_by_principal_id
describe_dashboard
describe_dashboard_permissions
describe_data_set
describe_data_set_permissions
describe_data_source
describe_data_source_permissions
describe_group
describe_iam_policy_assignment
describe_ingestion
describe_template
describe_template_alias
describe_template_permissions
describe_user
get_dashboard_embed_url
list_dashboards
list_dashboard_versions
list_data_sets
```

Cancels an ongoing ingestion of data into SPICE

Creates a dashboard from a template

Creates a dataset

Creates a data source

Creates an Amazon QuickSight group

Adds an Amazon QuickSight user to an Amazon QuickSight group

Creates an assignment with one specified IAM policy, identified by its Amazon Resource Name (ARN)

Creates and starts a new SPICE ingestion on a dataset

Any ingestions operating on tagged datasets inherit the same tags automatically for use in access control

Creates a template from an existing QuickSight analysis or template

Creates a template alias for a template

Deletes a dashboard

Deletes a dataset

Deletes the data source permanently

Removes a user group from Amazon QuickSight

Removes a user from a group so that the user is no longer a member of the group

Deletes an existing IAM policy assignment

Deletes a template

Deletes the item that the specified template alias points to

Deletes the Amazon QuickSight user that is associated with the identity of the AWS IAM user or role

Deletes a user identified by its principal ID

Provides a summary for a dashboard

Describes read and write permissions for a dashboard

Describes a dataset

Describes the permissions on a dataset

Describes a data source

Describes the resource permissions for a data source

Returns an Amazon QuickSight group’s description and Amazon Resource Name (ARN)

Describes an existing IAM policy assignment, as specified by the assignment name

Describes a SPICE ingestion

Describes a template’s metadata

Describes the template alias for a template

Describes read and write permissions on a template

Returns information about a user, given the user name

Generates a server-side embeddable URL and authorization code

Lists dashboards in an AWS account

Lists all the versions of the dashboards in the QuickSight subscription

Lists all of the datasets belonging to the current AWS account in an AWS Region
list_data_sources | Lists data sources in current AWS Region that belong to this AWS account
list_group_memberships | Lists member users in a group
list_groups | Lists all user groups in Amazon QuickSight
list_iam_policy_assignments | Lists IAM policy assignments in the current Amazon QuickSight account
list_iam_policy_assignments_for_user | Lists all the IAM policy assignments, including the Amazon Resource Names (ARNs) of the IAM policies assigned to the specified user and group or groups that the user belongs to
list_ingestions | Lists the history of SPICE ingestions for a dataset
list_tags_for_resource | Lists the tags assigned to a resource
list_template_aliases | Lists all the aliases of a template
list_templates | Lists all the templates in the current Amazon QuickSight account
list_template_versions | Lists all the versions of the templates in the current Amazon QuickSight account
list_user_groups | Lists the Amazon QuickSight groups that an Amazon QuickSight user is a member of
list_users | Returns a list of all the Amazon QuickSight users belonging to this account
register_user | Creates an Amazon QuickSight user, whose identity is associated with the AWS Identity and Access Management (IAM) identity or role specified in the request
tag_resource | Assigns one or more tags (key-value pairs) to the specified QuickSight resource
untag_resource | Removes a tag or tags from a resource
update_dashboard | Updates a dashboard in an AWS account
update_dashboard_permissions | Updates read and write permissions on a dashboard
update_dashboard_published_version | Updates the published version of a dashboard
update_data_set | Updates a dataset
update_data_set_permissions | Updates the permissions on a dataset
update_data_source | Updates a data source
update_data_source_permissions | Updates the permissions to a data source
update_group | Changes a group description
update_iam_policy_assignment | Updates an existing IAM policy assignment
update_template | Updates a template from an existing Amazon QuickSight analysis or another template
update_template_alias | Updates the template alias of a template
update_template_permissions | Updates the resource permissions for a template
update_user | Updates an Amazon QuickSight user

Examples

```r
svc <- quicksight()
svc$cancel_ingestion(
  Foo = 123
)
```
Index

accept_qualification_request, 24
activate_pipeline, 9
add_application_cloud_watch_logging_option, 21, 22
add_application_input, 21, 22
add_application_input_processing_configuration, 21, 22
add_application_output, 21, 22
add_application_reference_data_source, 21, 22
add_application_vpc_configuration, 22
add_instance_fleet, 12
add_instance_groups, 12
add_job_flow_steps, 12
add_tags, 9, 10, 12
add_tags_to_stream, 19
approve_assignment, 24
associate_qualification_with_worker, 24
athena, 2

batch_create_partition, 14
batch_delete_connection, 14
batch_delete_partition, 14
batch_delete_table, 14
batch_delete_table_version, 14
batch_get_crawlers, 14
batch_get_dev_endpoints, 15
batch_get_jobs, 15
batch_get_named_query, 3
batch_get_partition, 15
batch_get_query_execution, 3
batch_get_triggers, 15
batch_get_workflows, 15
batch_stop_job_run, 15
build_suggesters, 5
cancel_elasticsearch_service_software_update, 10
cancel_elasticsearch_domain, 5
cancel_ml_task_run, 15
cancel_steps, 12
calculate_transform, 15
cloudsearch, 4
cloudsearchdomain, 5
create_additional_assignments_for_hit, 24
create_application, 21, 22
create_connection, 15
create_data_set, 26
create_data_source, 26
create_database, 15
create_delivery_stream, 13
create_dev_endpoint, 15
create_domain, 5
create_elasticsearch_domain, 10
create_group, 26
create_group_membership, 26
create_ml_transform, 15
create_ml_task, 24
create_partition, 24
create_pipeline, 9
create_qualification_type, 24
create_script, 15
create_security_configuration, 12, 15
create_stream, 19
create_table, 15
create_template, 26
INDEX

create_template_alias, 26
create_trigger, 15
create_user_defined_function, 15
create_work_group, 3
create_worker_block, 24
create_workflow, 15
datapipeline, 7
deactivate_pipeline, 9
decrease_stream_retention_period, 19
define_analysis_scheme, 5
define_expression, 5
define_index_field, 5
define_suggester, 5
delete_analysis_scheme, 5
delete_application, 21, 22
delete_application_cloud_watch_logging_option, 21, 22
delete_application_input_processing_configuration, 21, 23
delete_application_output, 21, 23
delete_application_reference_data_source, 21, 23
delete_application_snapshot, 23
delete_application_vpc_configuration, 23
delete_classifier, 15
delete_cluster, 18
delete_connection, 15
delete_crawler, 15
delete_dashboard, 26
delete_data_set, 26
delete_data_source, 26
delete_database, 15
delete_delivery_stream, 13
delete_dev_endpoint, 15
delete_domain, 5
delete_elasticsearch_domain, 10
delete_elasticsearch_service_role, 10
delete_expression, 5
delete_group, 26
delete_group_membership, 26
delete_hit, 24
delete_iam_policy_assignment, 26
delete_index_field, 5
delete_job, 15
delete_ml_transform, 15
delete_named_query, 3
delete_partition, 15
delete_pipeline, 9
delete_qualification_type, 24
delete_resource_policy, 15
delete_security_configuration, 12, 15
delete_stream, 19
delete_suggester, 5
delete_table, 15
delete_table_version, 15
delete_template, 26
delete_template_alias, 26
delete_trigger, 15
delete_user, 26
delete_user_by_principal_id, 26
delete_user_defined_function, 15
delete_work_group, 3
delete_worker_block, 24
delete_workflow, 15
deregister_stream_consumer, 19
describe_analysis_schemes, 5
describe_application, 21, 23
describe_application_snapshot, 23
describe_availability_options, 5
describe_cluster, 12, 18
describe_cluster_operation, 18
describe_configuration, 18
describe_configuration_revision, 18
describe_dashboard, 26
describe_dashboard_permissions, 26
describe_data_set, 26
describe_data_set_permissions, 26
describe_data_source, 26
describe_data_source_permissions, 26
describe_delivery_stream, 13
describe_domain_endpoint_options, 5
describe_domains, 5
describe_elasticsearch_domain, 10
describe_elasticsearch_domain_config, 10
describe_elasticsearch_domains, 10
describe_elasticsearch_instance_type_limits, 10
describe_expressions, 5
describe_group, 26
describe_iam_policy_assignment, 26
describe_index_fields, 5
describe_ingestion, 26
describe_job_flows, 12
describe_limits, 19
describe_objects, 9
describe_pipelines, 9
describe_reserved_elasticsearch_instance_offerings, 10
describe_reserved_elasticsearch_instances, 10
describe_scaling_parameters, 5
describe_security_configuration, 12
describe_service_access_policies, 5
describe_step, 12
describe_stream, 19
describe_stream_consumer, 19
describe_stream_summary, 19
describe_suggesters, 5
describe_template, 26
describe_template_alias, 26
describe_template_permissions, 26
describe_user, 26
disable_enhanced_monitoring, 19
disassociate_qualification_from_worker, 24
discover_input_schema, 21, 23
elasticsearchservice, 9
emr, 11
enable_enhanced_monitoring, 19
evaluate_expression, 9

firehose, 12
get_account_balance, 24
get_assignment, 24
get_block_public_access_configuration, 12
get_bootstrap_brokers, 18
get_catalog_import_status, 15
get_classifier, 15
get_classifiers, 15
get_compatible_elasticsearch_versions, 10
get_connection, 15
get_connections, 15
get_crawler, 15
get_crawler_metrics, 15
get_crawlers, 15
get_dashboard_embed_url, 26
get_data_catalog_encryption_settings, 15
get_database, 15
get_databases, 15
get_dataflow_graph, 15
get_dev_endpoint, 16
get_dev_endpoints, 16
get_file_upload_url, 24
get_hit, 24
get_job, 16
get_job_bookmark, 16
get_job_run, 16
get_job_runs, 16
get_jobs, 16
get_mapping, 16
get_ml_task_run, 16
get_ml_task_runs, 16
get_ml_transform, 16
get_ml_transforms, 16
get_named_query, 3
get_partition, 16
get_partitions, 16
get_pipeline_definition, 9
get_plan, 16
get_qualification_score, 24
get_qualification_type, 24
get_query_execution, 3
get_query_results, 3
get_records, 20
get_resource_policy, 16
get_security_configuration, 16
get_security_configurations, 16
get_shard_iterator, 20
get_table, 16
get_table_version, 16
get_table_versions, 16
get_tables, 16
get_tags, 16
get_trigger, 16
get_triggers, 16
get_upgrade_history, 10
get_upgrade_status, 10
get_user_defined_function, 16
get_user_defined_functions, 16
get_work_group, 3
get_workflow, 16
get_workflow_run, 16
get_workflow_run_properties, 16
get_workflow_runs, 16

import_catalog_to_glue, 16

 glue, 14
INDEX

increase_stream_retention_period, 20
index_documents, 5
kafka, 17
kinesis, 19
kinesisanalytics, 20
kinesisanalyticsv2, 22
list_application_snapshots, 23
list_applications, 21, 23
list_assignments_for_hit, 24
list_bonus_payments, 24
list_bootstrap_actions, 12
list_cluster_operations, 18
list_clusters, 12, 18
list_configuration_revisions, 18
list_configurations, 18
list_crawlers, 16
list_dashboard_versions, 26
list_dashboards, 26
list_data_sets, 26
list_data_sources, 27
list_delivery_streams, 13
list_dev_endpoints, 16
list_domain_names, 5, 10
list_elasticsearch_instance_types, 10
list_elasticsearch_versions, 10
list_group_memberships, 27
list_groups, 27
list_hi_ts, 24
list_hi_ts_for_qualification_type, 24
list_iam_policy_assignments, 27
list_iam_policy_assignments_for_user, 27
list_ingestions, 27
list_instance_fleets, 12
list_instance_groups, 12
list_instances, 12
list_jobs, 16
list_named_queries, 3
list_nodes, 18
list_pipelines, 9
list_qualification_requests, 24
list_qualification_types, 24
list_query_executions, 3
list_review_policy_results_for_hit, 24
list_reviewable_hi_ts, 24
list_security_configurations, 12
list_shards, 20
list_steps, 12
list_stream_consumers, 20
list_streams, 20
list_tags, 10
list_tags_for_delivery_stream, 13
list_tags_for_resource, 3, 18, 21, 23, 27
list_tags_for_stream, 20
list_template_aliases, 27
list_template_versions, 27
list_templates, 27
list_triggers, 16
list_user_groups, 27
list_users, 27
list_work_groups, 3
list_worker_blocks, 24
list_workers_with_qualification_type, 24
list_workflows, 16
merge_shards, 20
modify_cluster, 12
modify_instance_fleet, 12
modify_instance_groups, 12
mturk, 23
notify_workers, 24
poll_for_task, 9
purchase_reserved_elasticsearch_instance_offering, 10
put_auto_scaling_policy, 12
put_block_public_access_configuration, 12
put_data_catalog_encryption_settings, 16
put_pipeline_definition, 9
put_record, 13, 20
put_record_batch, 13
put_records, 20
put_resource_policy, 16
put_workflow_run_properties, 16
query_objects, 9
quicksight, 25
register_stream_consumer, 20
register_user, 27
reject_assignment, 25
reject_qualification_request, 25
remove_auto_scaling_policy, 12
remove_tags, 9, 10, 12
remove_tags_from_stream, 20
report_task_progress, 9
report_task_runner_heartbeat, 9
reset_job_bookmark, 16
run_job_flow, 12
search, 6
search_tables, 16
send_bonus, 25
send_test_event_notification, 25
set_status, 9
set_task_status, 9
set_termination_protection, 12
set_visible_to_all_users, 12
split_shard, 20
start_application, 21, 23
start_crawler, 16
start_crawler_schedule, 16
start_delivery_stream_encryption, 13
start_elasticsearch_service_software_update, 10
start_export_labels_task_run, 16
start_import_labels_task_run, 16
start_ml_evaluation_task_run, 16
start_ml_labeling_set_generation_task_run, 17
start_query_execution, 3
start_stream_encryption, 20
start_trigger, 17
start_workflow_run, 17
stop_application, 21, 23
stop_crawler, 17
stop_crawler_schedule, 17
stop_delivery_stream_encryption, 13
stop_query_execution, 3
stop_stream_encryption, 20
stop_trigger, 17
suggest, 6
tag_delivery_stream, 13
tag_resource, 3, 17, 18, 21, 23, 27
terminate_job_flows, 12
untag_delivery_stream, 13
untag_resource, 3, 17, 18, 21, 23, 27
update_application, 21, 23
update_availability_options, 5
update_broker_count, 18
update_broker_storage, 18
update_classifier, 17
update_cluster_configuration, 18
update_connection, 17
update_crawler, 17
update_crawler_schedule, 17
update_dashboard, 27
update_dashboard_permissions, 27
update_dashboard_published_version, 27
update_data_set, 27
update_data_set_permissions, 27
update_data_source, 27
update_data_source_permissions, 27
update_database, 17
update_destination, 13
update_dev_endpoint, 17
update_domain_endpoint_options, 5
update_elasticsearch_domain_config, 10
update_expiration_for_hit, 25
update_group, 27
update_hit_review_status, 25
update_hit_type_of_hit, 25
update_iam_policy_assignment, 27
update_job, 17
update_ml_transform, 17
update_monitoring, 18
update_notification_settings, 25
update_partition, 17
update_qualification_type, 25
update_scaling_parameters, 5
update_service_access_policies, 5
update_table, 17
update_template, 27
update_template_alias, 27
update_template_permissions, 27
update_trigger, 17
update_user, 27
update_user_defined_function, 17
update_work_group, 3
update_workflow, 17
upgrade_elasticsearch_domain, 11
upload_documents, 6
validate_pipeline_definition, 9