Package ‘factset.analyticsapi.engines’

February 2, 2020

Title 'FactSet' Engines API Client
Version 3.0.1
Description Allow clients to fetch 'analytics' through API for Portfolio
'Analytics'('PA'), Style Performance Risk('SPAR') and 'Vault' products of
'FactSet'. Visit
<https://github.com/factset/analyticsapi-engines-r-sdk/tree/master/Engines>
for more information on the usage of package. Visit

URL https://github.com/factset/analyticsapi-engines-r-sdk

BugReports https://github.com/factset/analyticsapi-engines-r-sdk/issues

Depends R (>= 3.3)

Encoding UTF-8

License Apache License 2.0

LazyData true

Suggests testthat

Imports jsonlite, httr, R6, caTools, rlang

RoxygenNote 6.0.1.9000

NeedsCompilation no

Author Analytics API Support [aut],
Akshay Sheth [cre]

Maintainer Akshay Sheth <asheth@factset.com>

Repository CRAN

Date/Publication 2020-02-02 15:20:02 UTC

R topics documented:

AccountDirectories .............................................. 3
AccountsApi ......................................................... 4
ApiClient ............................................................ 7
ApiException ......................................................... 9
<table>
<thead>
<tr>
<th>Topic</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>ApiResponse</td>
<td>10</td>
</tr>
<tr>
<td>Calculation</td>
<td>11</td>
</tr>
<tr>
<td>CalculationsApi</td>
<td>12</td>
</tr>
<tr>
<td>CalculationStatus</td>
<td>23</td>
</tr>
<tr>
<td>CalculationStatusSummary</td>
<td>25</td>
</tr>
<tr>
<td>CalculationUnitStatus</td>
<td>26</td>
</tr>
<tr>
<td>Column</td>
<td>28</td>
</tr>
<tr>
<td>ColumnsApi</td>
<td>29</td>
</tr>
<tr>
<td>ColumnStatistic</td>
<td>35</td>
</tr>
<tr>
<td>ColumnStatisticsApi</td>
<td>36</td>
</tr>
<tr>
<td>ColumnSummary</td>
<td>39</td>
</tr>
<tr>
<td>ComponentAccount</td>
<td>40</td>
</tr>
<tr>
<td>ComponentBenchmark</td>
<td>42</td>
</tr>
<tr>
<td>ComponentsApi</td>
<td>43</td>
</tr>
<tr>
<td>ComponentSummary</td>
<td>57</td>
</tr>
<tr>
<td>ConfigurationAccount</td>
<td>58</td>
</tr>
<tr>
<td>ConfigurationsApi</td>
<td>60</td>
</tr>
<tr>
<td>CurrenciesApi</td>
<td>66</td>
</tr>
<tr>
<td>Currency</td>
<td>69</td>
</tr>
<tr>
<td>DateParametersSummary</td>
<td>70</td>
</tr>
<tr>
<td>DatesApi</td>
<td>72</td>
</tr>
<tr>
<td>DocumentDirectories</td>
<td>78</td>
</tr>
<tr>
<td>DocumentsApi</td>
<td>80</td>
</tr>
<tr>
<td>FrequenciesApi</td>
<td>88</td>
</tr>
<tr>
<td>Frequency</td>
<td>96</td>
</tr>
<tr>
<td>Group</td>
<td>97</td>
</tr>
<tr>
<td>GroupsApi</td>
<td>98</td>
</tr>
<tr>
<td>PACalculationColumn</td>
<td>101</td>
</tr>
<tr>
<td>PACalculationGroup</td>
<td>102</td>
</tr>
<tr>
<td>PACalculationParameters</td>
<td>104</td>
</tr>
<tr>
<td>PACOMPONENT</td>
<td>105</td>
</tr>
<tr>
<td>PADateParameters</td>
<td>107</td>
</tr>
<tr>
<td>PAIdentifier</td>
<td>108</td>
</tr>
<tr>
<td>SPARCalculationParameters</td>
<td>109</td>
</tr>
<tr>
<td>SPARDateParameters</td>
<td>110</td>
</tr>
<tr>
<td>SPARIdentifier</td>
<td>112</td>
</tr>
<tr>
<td>UtilityApi</td>
<td>113</td>
</tr>
<tr>
<td>VaultCalculationParameters</td>
<td>116</td>
</tr>
<tr>
<td>VaultComponent</td>
<td>117</td>
</tr>
<tr>
<td>VaultConfiguration</td>
<td>119</td>
</tr>
<tr>
<td>VaultConfigurationSummary</td>
<td>120</td>
</tr>
<tr>
<td>VaultDateParameters</td>
<td>121</td>
</tr>
<tr>
<td>VaultIdentifier</td>
<td>122</td>
</tr>
</tbody>
</table>

**Index** 124
**AccountDirectories**

---

### Description

AccountDirectories Class

### Format

An R6Class generator object

### Public fields

- accounts list(character) [optional]
- directories list(character) [optional]

### Methods

**Public methods:**

- `AccountDirectories$new()`
- `AccountDirectories$toJSON()`
- `AccountDirectories$fromJSON()`
- `AccountDirectories$toJSONString()`
- `AccountDirectories$fromJSONString()`
- `AccountDirectories$clone()`

**Method new():**

*Usage:*

AccountDirectories$new(accounts = NULL, directories = NULL, ...)

**Method toJSON():**

*Usage:*

AccountDirectories$toJSON()

**Method fromJSON():**

*Usage:*

AccountDirectories$fromJSON(AccountDirectoriesJson)

**Method toJSONString():**

*Usage:*

AccountDirectories$toJSONString()

**Method fromJSONString():**

*Usage:*

AccountDirectories$fromJSONString(AccountDirectoriesJson)
Method clone(): The objects of this class are cloneable with this method.

Usage:
AccountDirectories$clone(deep = FALSE)

Arguments:
deep Whether to make a deep clone.
AccountsApi

- status code: 401 | Missing or invalid authentication
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet's request key header.

- status code: 403 | User is forbidden with current credentials
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet's request key header.

- status code: 404 | Path not found
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet's request key header.

- status code: 406 | Unsupported Accept header. Header needs to be set to application/json
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet's request key header.

- status code: 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

• status code: 503 | Request timed out. Retry the request in some time
• response headers:

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

Public fields

apiClient  Handles the client-server communication.

Methods

Public methods:

• AccountsApi$new()
• AccountsApi$GetAccounts()
• AccountsApi$GetAccountsWithHttpInfo()
• AccountsApi$clone()

Method new():

Usage:
AccountsApi$new(apiClient)

Method GetAccounts():

Usage:
AccountsApi$GetAccounts(path, ...)

Method GetAccountsWithHttpInfo():

Usage:
AccountsApi$GetAccountsWithHttpInfo(path, ...)

Method clone(): The objects of this class are cloneable with this method.

Usage:
AccountsApi$clone(deep = FALSE)

Arguments:

deep  Whether to make a deep clone.
## Examples

```
### Not run:  
#################### GetAccounts ####################
library(factset.analyticsapi.engines)
var.path <- 'path_example' # character | The directory to get the accounts and sub-directories in

#Get accounts and sub-directories in a directory
api.instance <- AccountsApi$new()

#Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>';
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>';

result <- tryCatch(
    api.instance$GetAccounts(var.path),
    ApiException = function(ex) ex
)
# In case of error, print the error object
if(!is.null(result$ApiException)) {
    cat(result$ApiException$toString())
} else {
    # deserialized response object
    response.object <- result$content
    # response headers
    response.headers <- result$response$headers
    # response status code
    response.status.code <- result$response$status_code
}
```

## End(Not run)

---

### Description

ApiClient Class

### Format

An R6Class generator object
ApiClient

Details
ApiClient Class
Generic API client for OpenAPI client library builds. OpenAPI generic API client. This client handles the client-server communication, and is invariant across implementations. Specifics of the methods and models for each application are generated from the OpenAPI Generator templates.

NOTE: This class is auto generated by OpenAPI Generator (https://openapi-generator.tech). Ref: https://openapi-generator.tech Do not edit the class manually.

Public fields
basePath Base url
userAgent Default user agent
username Username for HTTP basic authentication
password Password for HTTP basic authentication
timeout Default timeout in seconds

Methods
Public methods:
• ApiClient$new()
• ApiClient$CallApi()
• ApiClient$deserialize()
• ApiClient$deserializeObj()
• ApiClient$clone()

Method new():
Usage:
ApiClient$new(  
  basePath = NULL,  
  userAgent = NULL,  
  defaultHeaders = NULL,  
  username = NULL,  
  password = NULL,  
  apiKeys = NULL,  
  accessToken = NULL,  
  timeout = NULL  
)

Method CallApi():
Usage:
ApiClient$CallApi(url, method, queryParams, headerParams, body, ...)

Method deserialize():
Usage:
ApiClient$deserialize(resp, returnType, pkgEnv)
**Method** deserializeObj():

*Usage:*

```r
ApiClient$deserializeObj(obj, returnType, pkgEnv)
```

**Method** clone(): The objects of this class are cloneable with this method.

*Usage:*

```r
ApiClient$clone(deep = FALSE)
```

*Arguments:*

- deep Whether to make a deep clone.

---

**ApiException**

**Description**

ApiException Class

**Format**

An R6Class generator object

**Public fields**

- `status` Status of the ApiException
- `reason` Reason of the ApiException
- `body` Body of the http response
- `headers` Headers of the http response

**Methods**

**Public methods:**

- `ApiException$new()`
- `ApiException$toString()`
- `ApiException$clone()`

**Method** new():

*Usage:*

```r
ApiException$new(status = NULL, reason = NULL, http_response = NULL)
```

**Method** toString():

*Usage:*

```r
ApiException$toString()
```

**Method** clone(): The objects of this class are cloneable with this method.
ApiResponse

Usage:
ApiResponse$clone(deep = FALSE)

Arguments:
deep Whether to make a deep clone.

Description
ApiResponse Class

Format
An R6Class generator object

Public fields
content The deserialized response body.
response The raw response from the endpoint.

Methods
Public methods:
- ApiResponse$new()
- ApiResponse$clone()

Method new():
Usage:
ApiResponse$new(content, response)

Method clone(): The objects of this class are cloneable with this method.
Usage:
ApiResponse$clone(deep = FALSE)

Arguments:
deep Whether to make a deep clone.
Description

Calculation Class

Format

An R6Class generator object

Public fields

- pa named list( PACalculationParameters ) [optional]
- spar named list( SPARCalculationParameters ) [optional]
- vault named list( VaultCalculationParameters ) [optional]

Methods

Public methods:

- Calculation$new()
- Calculation$toJSON()
- Calculation$fromJSON()
- Calculation$toJSONString()
- Calculation$fromJSONString()
- Calculation$clone()

Method new():

Usage:
Calculation$new(pa = NULL, spar = NULL, vault = NULL, ...)

Method toJSON():

Usage:
Calculation$toJSON()

Method fromJSON():

Usage:
Calculation$fromJSON(CalculationJson)

Method toJSONString():

Usage:
Calculation$toJSONString()

Method fromJSONString():


Usage:
Calculation$fromJSONString(CalculationJson)

Method clone(): The objects of this class are cloneable with this method.
Usage:
Calculation$clone(deep = FALSE)

Arguments:
depth Whether to make a deep clone.

Description
factsset.analyticsapi.engines.Calculations

Format
An R6Class generator object

Methods

CancelCalculationById  Cancel calculation by id  This is the endpoint to cancel a previously submitted calculation request. Instead of doing a GET on the getCalculationById URL, cancel the calculation by doing a DELETE. All individual calculation units within the calculation will be canceled if they have not already finished.

@param id character
• On encountering errors, an error of subclass ApiException will be thrown.
• status code : 204 | Expected response, request was cancelled successfully.
• response headers :
  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet's request key header.

• status code : 400 | Invalid identifier provided.
• response headers :
  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet's request key header.
CalculationsApi

- status code: 401 | Missing or invalid authentication.
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet's request key header.

- status code: 403 | User is forbidden with current credentials.
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet's request key header.

- status code: 404 | There was no request for the identifier provided, or the request was already canceled for the provided identifier.
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet's request key header.

- status code: 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting.
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet's request key header.

- status code: 503 | Request timed out. Retry the request in sometime.
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
CalculationsApi

**GetCalculationStatusById**  
*Get calculation status by id*  
This is the endpoint to check on the progress of a previous calculation request. Response body contains status information of the entire request and each individual calculation unit.

- @param id character
- @returnType CalculationStatus

- On encountering errors, an error of subclass ApiException will be thrown.
- status code : 200 | Expected response, returns status information of the entire calculation and each individual calculation unit.
- return type : CalculationStatus
- response headers :

  - X-RateLimit-Remaining Number of requests left for the time window.
  - Cache-Control Standard HTTP header. Header will specify max-age in seconds. Polling can be adjusted based on the max-age value.
  - X-FactSet_Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit Number of allowed requests for the time window.
  - X-DataDirect-Request-Key FactSet’s request key header.

- status code : 400 | Invalid identifier provided.
- response headers :

  - X-RateLimit-Remaining Number of requests left for the time window.
  - Cache-Control Standard HTTP header. Header will specify max-age in seconds. Polling can be adjusted based on the max-age value.
  - X-FactSet_Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit Number of allowed requests for the time window.
  - X-DataDirect-Request-Key FactSet’s request key header.

- status code : 401 | Missing or invalid authentication.
- response headers :

  - X-RateLimit-Remaining Number of requests left for the time window.
  - Cache-Control Standard HTTP header. Header will specify max-age in seconds. Polling can be adjusted based on the max-age value.
  - X-FactSet_Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit Number of allowed requests for the time window.
  - X-DataDirect-Request-Key FactSet’s request key header.
• status code : 403 | User is forbidden with current credentials.

response headers:

X-RateLimit-Remaining Number of requests left for the time window.
Cache-Control Standard HTTP header. Header will specify max-age in seconds. Polling can be adjusted based on the max-age value.
X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset Number of seconds remaining till rate limit resets.
X-RateLimit-Limit Number of allowed requests for the time window.
X-DataDirect-Request-Key FactSet's request key header.

• status code : 404 | Provided identifier was not a request, or the request was cancelled.

response headers:

X-RateLimit-Remaining Number of requests left for the time window.
Cache-Control Standard HTTP header. Header will specify max-age in seconds. Polling can be adjusted based on the max-age value.
X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset Number of seconds remaining till rate limit resets.
X-RateLimit-Limit Number of allowed requests for the time window.
X-DataDirect-Request-Key FactSet's request key header.

• status code : 406 | Unsupported Accept header. Header needs to be set to application/json.

response headers:

X-RateLimit-Remaining Number of requests left for the time window.
Cache-Control Standard HTTP header. Header will specify max-age in seconds. Polling can be adjusted based on the max-age value.
X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset Number of seconds remaining till rate limit resets.
X-RateLimit-Limit Number of allowed requests for the time window.
X-DataDirect-Request-Key FactSet's request key header.

• status code : 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting.

response headers:

X-RateLimit-Remaining Number of requests left for the time window.
Cache-Control Standard HTTP header. Header will specify max-age in seconds. Polling can be adjusted based on the max-age value.
X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset Number of seconds remaining till rate limit resets.
X-RateLimit-Limit Number of allowed requests for the time window.
X-DataDirect-Request-Key FactSet's request key header.

• status code : 503 | Request timed out. Retry the request in sometime.

response headers:
GetCalculationStatusSummaries  Get all calculation statuses  This endpoint returns all active calculation requests.

@returnType named list( CalculationStatusSummary )

• On encountering errors, an error of subclass ApiException will be thrown.
• status code : 200 | List of active calculation requests.
• return type : map(CalculationStatusSummary)
• response headers :

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

• status code : 401 | Missing or invalid authentication.
• response headers :

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

• status code : 403 | User is forbidden with current credentials.
• response headers :

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

• status code : 406 | Unsupported Accept header. Header needs to be set to application/json.
• response headers :

X-RateLimit-Remaining  Number of requests left for the time window.
CalculationsApi

X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset Number of seconds remaining till rate limit resets.
X-RateLimit-Limit Number of allowed requests for the time window.
X-DataDirect-Request-Key FactSet's request key header.

• status code : 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting.
  • response headers :
    X-RateLimit-Remaining Number of requests left for the time window.
    X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
    X-RateLimit-Reset Number of seconds remaining till rate limit resets.
    X-RateLimit-Limit Number of allowed requests for the time window.
    X-DataDirect-Request-Key FactSet's request key header.

• status code : 503 | Request timed out. Retry the request in sometime.
  • response headers :
    X-RateLimit-Remaining Number of requests left for the time window.
    X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
    X-RateLimit-Reset Number of seconds remaining till rate limit resets.
    X-RateLimit-Limit Number of allowed requests for the time window.
    X-DataDirect-Request-Key FactSet's request key header.

RunCalculation Run calculation This endpoint creates a new calculation and runs the set of calculation units specified in the POST body. This must be used first before get status or cancelling endpoints with a calculation id. A successful response will contain the URL to check the status of the calculation request. Remarks: * Maximum 25 points allowed per calculation and maximum 500 points allowed across all simultaneous calculations. (Refer API documentation for more information) * Any settings in POST body will act as a one-time override over the settings saved in the PA/SPAR/Vault template.

  @param calculation Calculation
  • On encountering errors, an error of subclass ApiException will be thrown.
  • status code : 202 | Expected response, contains the URL in the Location header to check the status of the calculation.
  • response headers :
    X-FactSet-Api-Points-Limit Maximum points limit across all batches.
    X-RateLimit-Remaining Number of requests left for the time window.
    X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
    X-FactSet-Api-Points-Remaining Number of points remaining till points limit reached.
    X-RateLimit-Reset Number of seconds remaining till rate limit resets.
    X-RateLimit-Limit Number of allowed requests for the time window.
    Location URL to check status of the request.
    X-DataDirect-Request-Key FactSet's request key header.
• status code: 400 | Invalid POST body.
• response headers:
  X-FactSet-Api-Points-Limit: Maximum points limit across all batches.
  X-RateLimit-Remaining: Number of requests left for the time window.
  X-FactSet-API-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-FactSet-Api-Points-Remaining: Number of points remaining till points limit reached.
  X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit: Number of allowed requests for the time window.
  Location: URL to check status of the request.
  X-DataDirect-Request-Key: FactSet's request key header.

• status code: 401 | Missing or invalid authentication.
• response headers:
  X-FactSet-Api-Points-Limit: Maximum points limit across all batches.
  X-RateLimit-Remaining: Number of requests left for the time window.
  X-FactSet-API-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-FactSet-Api-Points-Remaining: Number of points remaining till points limit reached.
  X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit: Number of allowed requests for the time window.
  Location: URL to check status of the request.
  X-DataDirect-Request-Key: FactSet's request key header.

• status code: 403 | User is forbidden with current credentials.
• response headers:
  X-FactSet-Api-Points-Limit: Maximum points limit across all batches.
  X-RateLimit-Remaining: Number of requests left for the time window.
  X-FactSet-API-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-FactSet-Api-Points-Remaining: Number of points remaining till points limit reached.
  X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit: Number of allowed requests for the time window.
  Location: URL to check status of the request.
  X-DataDirect-Request-Key: FactSet's request key header.

• status code: 415 | Missing/Invalid Content-Type header. Header needs to be set to application/json.
• response headers:
  X-FactSet-Api-Points-Limit: Maximum points limit across all batches.
  X-RateLimit-Remaining: Number of requests left for the time window.
  X-FactSet-API-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
• status code: 429 | Rate limit reached. Cancel older requests using Cancel Calculation endpoint or wait for older requests to finish / expire.

• response headers:

X-FactSet-Api-Points-Limit Maximum points limit across all batches.
X-RateLimit-Remaining Number of requests left for the time window.
X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-FactSet-Api-Points-Remaining Number of points remaining till points limit reached.
X-RateLimit-Reset Number of seconds remaining till rate limit resets.
X-RateLimit-Limit Number of allowed requests for the time window.
Location URL to check status of the request.
X-DataDirect-Request-Key FactSet’s request key header.

• status code: 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting.

• response headers:

X-FactSet-Api-Points-Limit Maximum points limit across all batches.
X-RateLimit-Remaining Number of requests left for the time window.
X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-FactSet-Api-Points-Remaining Number of points remaining till points limit reached.
X-RateLimit-Reset Number of seconds remaining till rate limit resets.
X-RateLimit-Limit Number of allowed requests for the time window.
Location URL to check status of the request.
X-DataDirect-Request-Key FactSet’s request key header.

• status code: 503 | Request timed out. Retry the request in sometime.

• response headers:

X-FactSet-Api-Points-Limit Maximum points limit across all batches.
X-RateLimit-Remaining Number of requests left for the time window.
X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-FactSet-Api-Points-Remaining Number of points remaining till points limit reached.
X-RateLimit-Reset Number of seconds remaining till rate limit resets.
X-RateLimit-Limit Number of allowed requests for the time window.
Location URL to check status of the request.
X-DataDirect-Request-Key FactSet’s request key header.
Public fields

apiClient  Handles the client-server communication.

Methods

Public methods:

• CalculationsApi$new()
• CalculationsApi$cancelCalculationById()  
• CalculationsApi$cancelCalculationByIdWithHttpInfo()  
• CalculationsApi$getCalculationStatusById()  
• CalculationsApi$getCalculationStatusByIdWithHttpInfo()  
• CalculationsApi$getCalculationStatusSummaries()  
• CalculationsApi$getCalculationStatusSummariesWithHttpInfo()  
• CalculationsApi$runCalculation()  
• CalculationsApi$clone()

Method new():
Usage:
CalculationsApi$new(apiClient)

Method CancelCalculationById():
Usage:
CalculationsApi$cancelCalculationById(id, ...)

Method CancelCalculationByIdWithHttpInfo():
Usage:
CalculationsApi$cancelCalculationByIdWithHttpInfo(id, ...)

Method GetCalculationStatusById():
Usage:
CalculationsApi$getCalculationStatusById(id, ...)

Method GetCalculationStatusByIdWithHttpInfo():
Usage:
CalculationsApi$getCalculationStatusByIdWithHttpInfo(id, ...)

Method GetCalculationStatusSummaries():
Usage:
CalculationsApi$getCalculationStatusSummaries(...)

Method GetCalculationStatusSummariesWithHttpInfo():
Usage:
CalculationsApi$getCalculationStatusSummariesWithHttpInfo(...
Method RunCalculation():
  Usage:
  CalculationsApi$RunCalculation(calculation = NULL, ...)

Method RunCalculationWithHttpInfo():
  Usage:
  CalculationsApi$RunCalculationWithHttpInfo(calculation = NULL, ...)

Method clone(): The objects of this class are cloneable with this method.
  Usage:
  CalculationsApi$clone(deep = FALSE)
  Arguments:
  deep Whether to make a deep clone.

Examples

```r
## Not run:
library(factset.analyticsapi.engines)
var.id <- 'id_example' # character | From url, provided from the location
  # header in the Run Multiple Calculations endpoint.

#Cancel calculation by id
api.instance <- CalculationsApi$new()

#Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>,'
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>,'

result <- tryCatch(
  api.instance$CancelCalculationByVar.id(var.id),
  ApiException = function(ex) ex
)
# In case of error, print the error object
if(!is.null(result$ApiException)) {
  cat(result$ApiException$toString())
} else {
  # response headers
  response.headers <- result$response$headers
  # response status code
  response.status.code <- result$response$status_code
}
```

GetCalculationStatusByVar.id

```r
library(factset.analyticsapi.engines)
```
var.id <- 'id_example' # character | From url, provided from the location
   # header in the Run Multiple Calculations endpoint.

#Get calculation status by id
api.instance <- CalculationsApi$new()

#Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>'
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>'

result <- tryCatch(
   api.instance$GetCalculationStatusById(var.id),
   ApiException = function(ex) ex
)
# In case of error, print the error object
if(!is.null(result$ApiException)) {
   cat(result$ApiException$toString())
} else {
   # deserialized response object
   response.object <- result$content
   # response headers
   response.headers <- result$response$headers
   # response status code
   response.status.code <- result$response$status_code
}

########################################################################

library(factset.analyticsapi.engines)

#Get all calculation statuses
api.instance <- CalculationsApi$new()

#Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>'
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>'

result <- tryCatch(
   api.instance$GetCalculationStatusSummaries(),
   ApiException = function(ex) ex
)
# In case of error, print the error object
if(!is.null(result$ApiException)) {
   cat(result$ApiException$toString())
} else {
   # deserialized response object
   response.object <- result$content
   # response headers
response.headers <- result$response$headers
# response status code
response.status.code <- result$response$status_code
}

######################### RunCalculation #########################

library(factset.analyticsapi.engines)
var.calculation <- Calculation$new() # Calculation |

#Run calculation
api.instance <- CalculationsApi$new()

#Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '</user-serial>';
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '</api_key>';

result <- tryCatch(
  api.instance$RunCalculation(calculation=var.calculation),
  ApiException = function(ex) ex
)

# In case of error, print the error object
if(!is.null(result$ApiException)) {
  cat(result$ApiException$toString())
} else {
  # response headers
  response.headers <- result$response$headers
  # response status code
  response.status.code <- result$response$status_code
}

## End(Not run)

---

**CalculationStatus**

**CalculationStatus**

**Description**

CalculationStatus Class

**Format**

An R6Class generator object
Public fields

- status character [optional]
- points integer [optional]
- pa named list( CalculationUnitStatus ) [optional]
- spar named list( CalculationUnitStatus ) [optional]
- vault named list( CalculationUnitStatus ) [optional]

Methods

Public methods:
- CalculationStatus$new()
- CalculationStatus$toJSON()
- CalculationStatus$fromJSON()
- CalculationStatus$toJSONString()
- CalculationStatus$fromJSONString()
- CalculationStatus$clone()

Method new():
Usage:
CalculationStatus$new(
  status = NULL,
  points = NULL,
  pa = NULL,
  spar = NULL,
  vault = NULL,
  ...
)

Method toJSON():
Usage:
CalculationStatus$toJSON()

Method fromJSON():
Usage:
CalculationStatus$fromJSON(CalculationStatusJson)

Method toJSONString():
Usage:
CalculationStatus$toJSONString()

Method fromJSONString():
Usage:
CalculationStatus$fromJSONString(CalculationStatusJson)

Method clone(): The objects of this class are cloneable with this method.
CalculationStatusSummary

Usage:
CalculationStatusSummary$clone(deep = FALSE)

Arguments:
depth Whether to make a deep clone.

Description
CalculationStatusSummary Class

Format
An R6Class generator object

Public fields
status character [optional]
points integer [optional]
requestTime character [optional]
lastPollTime character [optional]

Methods
Public methods:
- CalculationStatusSummary$new()
- CalculationStatusSummary$toJSON()
- CalculationStatusSummary$fromJSON()
- CalculationStatusSummary$toJSONString()
- CalculationStatusSummary$fromJSONString()
- CalculationStatusSummary$clone()

Method new():
Usage:
CalculationStatusSummary$new(
    status = NULL,
    points = NULL,
    requestTime = NULL,
    lastPollTime = NULL,
    ...)

Method toJSON():
Usage:
CalculationStatusSummary$toJSON()

Method fromJSON():
Usage:
CalculationStatusSummary$fromJSON(CalculationStatusSummaryJson)

Method toJSONString():
Usage:
CalculationStatusSummary$toJSONString()

Method fromJSONString():
Usage:
CalculationStatusSummary$fromJSONString(CalculationStatusSummaryJson)

Method clone(): The objects of this class are cloneable with this method.
Usage:
CalculationStatusSummary$clone(deep = FALSE)
Arguments:
deep  Whether to make a deep clone.

CalculationUnitStatus  CalculationUnitStatus

Description
CalculationUnitStatus Class

Format
An R6Class generator object

Public fields
status  character [optional]
points  integer [optional]
error   character [optional]
result  character [optional]
Methods

Public methods:

• CalculationUnitStatus$new()
• CalculationUnitStatus$toJSON()
• CalculationUnitStatus$fromJSON()
• CalculationUnitStatus$toJSONString()
• CalculationUnitStatus$fromJSONString()
• CalculationUnitStatus$clone()

Method new():

Usage:
CalculationUnitStatus$new(
  status = NULL,
  points = NULL,
  error = NULL,
  result = NULL,
  ...
)

Method toJSON():

Usage:
CalculationUnitStatus$toJSON()

Method fromJSON():

Usage:
CalculationUnitStatus$fromJSON(CalculationUnitStatusJson)

Method toJSONString():

Usage:
CalculationUnitStatus$toJSONString()

Method fromJSONString():

Usage:
CalculationUnitStatus$fromJSONString(CalculationUnitStatusJson)

Method clone(): The objects of this class are cloneable with this method.

Usage:
CalculationUnitStatus$clone(deep = FALSE)

Arguments:

deep Whether to make a deep clone.
Description

Column Class

Format

An R6Class generator object

Public fields

defaultstatisticsids list(character) [optional]
name character [optional]
directory character [optional]
category character [optional]

Methods

Public methods:

- Column$new()
- Column$toJSON()
- Column$fromJSON()
- Column$toJSONString()
- Column$fromJSONString()
- Column$clone()

Method new():

Usage:
Column$new(
  defaultstatisticsids = NULL,
  name = NULL,
  directory = NULL,
  category = NULL,
  ...
)

Method toJSON():

Usage:
Column$toJSON()

Method fromJSON():

Usage:
ColumnsApi

Column$fromJSON(ColumnJson)

Method toJSONString():
  Usage:
  Column$toJSONString()

Method fromJSONString():
  Usage:
  Column$fromJSONString(ColumnJson)

Method clone(): The objects of this class are cloneable with this method.
  Usage:
  Column$clone(deep = FALSE)
  Arguments:
  deep Whether to make a deep clone.

ColumnsApi  Columns operations

Description
  factset.analyticsapi.engines.Columns

Format
  An R6Class generator object

Methods
  GetPAColumnById  Get PA column settings  This endpoint returns the default settings of a PA column.
      @param id character
      •  @returnType Column
        •  On encountering errors, an error of subclass ApiException will be thrown.
        •  status code : 200 | Expected response, returns settings of a PA column.
        •  return type : Column
        •  response headers :
          X-RateLimit-Remaining  Number of requests left for the time window.
          X-FactSet-API-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
          X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
          X-RateLimit-Limit  Number of allowed requests for the time window.
          Age  Standard HTTP header. Header will specify the age of columns list cached response.
          X-DataDirect-Request-Key  FactSet's request key header.
• status code: 400 | Invalid column Id.
  • response headers:
    X-RateLimit-Remaining: Number of requests left for the time window.
    X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
    X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
    X-RateLimit-Limit: Number of allowed requests for the time window.
    X-DataDirect-Request-Key: FactSet's request key header.

• status code: 401 | Missing or invalid authentication.
  • response headers:
    X-RateLimit-Remaining: Number of requests left for the time window.
    X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
    X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
    X-RateLimit-Limit: Number of allowed requests for the time window.
    X-DataDirect-Request-Key: FactSet's request key header.

• status code: 403 | User is forbidden with current credentials.
  • response headers:
    X-RateLimit-Remaining: Number of requests left for the time window.
    X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
    X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
    X-RateLimit-Limit: Number of allowed requests for the time window.
    X-DataDirect-Request-Key: FactSet's request key header.

• status code: 404 | Column not found.
  • response headers:
    X-RateLimit-Remaining: Number of requests left for the time window.
    X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
    X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
    X-RateLimit-Limit: Number of allowed requests for the time window.
    X-DataDirect-Request-Key: FactSet's request key header.

• status code: 406 | Unsupported Accept header. Header needs to be set to application/json.
  • response headers:
    X-RateLimit-Remaining: Number of requests left for the time window.
    X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
    X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
    X-RateLimit-Limit: Number of allowed requests for the time window.
    X-DataDirect-Request-Key: FactSet's request key header.
- status code : 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting.
- response headers :

  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet’s request key header.

- status code : 503 | Request timed out. Retry the request in sometime.
- response headers :

  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet’s request key header.

**GetPAColumns** *Get PA columns* This endpoint lists all the PA columns that can be applied to a calculation.

  @param name character
  @param category character
  @param directory character
  @returnType named list( ColumnSummary )

- On encountering errors, an error of subclass ApiException will be thrown.
- status code : 200 | Expected response, returns a list of PA columns
- return type : map(ColumnSummary)
- response headers :

  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  Age Standard HTTP header. Header will specify the age of columns list cached response.
  X-DataDirect-Request-Key FactSet’s request key header.

- status code : 401 | Missing or invalid authentication.
- response headers :

  X-RateLimit-Remaining Number of requests left for the time window.
- status code : 403 | User is forbidden with current credentials
  - response headers :
    - X-RateLimit-Remaining Number of requests left for the time window.
    - X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
    - X-RateLimit-Reset Number of seconds remaining till rate limit resets.
    - X-RateLimit-Limit Number of allowed requests for the time window.
    - X-DataDirect-Request-Key FactSet’s request key header.

- status code : 406 | Unsupported Accept header. Header needs to be set to application/json.
  - response headers :
    - X-RateLimit-Remaining Number of requests left for the time window.
    - X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
    - X-RateLimit-Reset Number of seconds remaining till rate limit resets.
    - X-RateLimit-Limit Number of allowed requests for the time window.
    - X-DataDirect-Request-Key FactSet’s request key header.

- status code : 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting
  - response headers :
    - X-RateLimit-Remaining Number of requests left for the time window.
    - X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
    - X-RateLimit-Reset Number of seconds remaining till rate limit resets.
    - X-RateLimit-Limit Number of allowed requests for the time window.
    - X-DataDirect-Request-Key FactSet’s request key header.

- status code : 503 | Request timed out. Retry the request in sometime.
  - response headers :
    - X-RateLimit-Remaining Number of requests left for the time window.
    - X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
    - X-RateLimit-Reset Number of seconds remaining till rate limit resets.
    - X-RateLimit-Limit Number of allowed requests for the time window.
    - X-DataDirect-Request-Key FactSet’s request key header.
Public fields

apiClient  Handles the client-server communication.

Methods

Public methods:

• ColumnsApi$new()
• ColumnsApi$GetPAColumnById()
• ColumnsApi$GetPAColumnByIdWithHttpInfo()
• ColumnsApi$GetPAColumns()
• ColumnsApi$GetPAColumnsWithHttpInfo()
• ColumnsApi$clone()

Method new():

Usage:
ColumnsApi$new(apiClient)

Method GetPAColumnById():

Usage:
ColumnsApi$GetPAColumnById(id, ...)

Method GetPAColumnByIdWithHttpInfo():

Usage:
ColumnsApi$GetPAColumnByIdWithHttpInfo(id, ...)

Method GetPAColumns():

Usage:
ColumnsApi$GetPAColumns(name = NULL, category = NULL, directory = NULL, ...)

Method GetPAColumnsWithHttpInfo():

Usage:
ColumnsApi$GetPAColumnsWithHttpInfo(
    name = NULL,
    category = NULL,
    directory = NULL,
    ...
)

Method clone(): The objects of this class are cloneable with this method.

Usage:
ColumnsApi$clone(deep = FALSE)

Arguments:

deep  Whether to make a deep clone.
Examples

```r
## Not run:  
library(factset.analyticsapi.engines)  
var.id <- 'id_example' # character | Unique identifier for a column

#Get PA column settings  
api.instance < ColumnsApi$new()

#Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username < 'user-serial';
# provide your api key generated using the developer portal
api.instance$apiClient$password < 'api_key';

result <- tryCatch(
    api.instance$GetPAColumnById(var.id),
    ApiException = function(ex) ex
)

# In case of error, print the error object
if(!is.null(result$ApiException)) {
    cat(result$ApiException$toString())
} else {

# deserialized response object
response.object <- result$content
# response headers
response.headers <- result$response$headers
# response status code
response.status.code <- result$response$status_code
}

## GetPAColumns

library(factset.analyticsapi.engines)  
var.name <- 'name_example' # character |
var.category <- 'category_example' # character |
var.directory <- 'directory_example' # character |

#Get PA columns  
api.instance < ColumnsApi$new()

#Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username < 'user-serial';
# provide your api key generated using the developer portal
api.instance$apiClient$password < 'api_key';

result <- tryCatch(
    api.instance$GetPAColumns(name=var.name, category=var.category,
                              directory=var.directory),
    ApiException = function(ex) ex
)
```

```
ApiException = function(ex) ex
)

# In case of error, print the error object
if(!is.null(result$ApiException)) {
  cat(result$ApiException$toString())
} else {
  # deserialized response object
  response.object <- result$content
  # response headers
  response.headers <- result$response$headers
  # response status code
  response.status.code <- result$response$status_code
}

## End(Not run)

ColumnStatistic

**Description**

ColumnStatistic Class

**Format**

An R6Class generator object

**Public fields**

- name character [optional]

**Methods**

**Public methods:**

- ColumnStatistic$new()
- ColumnStatistic$toJSON()
- ColumnStatistic$fromJSON()
- ColumnStatistic$toJSONString()
- ColumnStatistic$fromJSONString()
- ColumnStatistic$clone()

**Method** new():

*Usage:*

ColumnStatistic$new(name = NULL, ...)

**Method** toJSON():


Usage:
ColumnStatistic$toJSON()

Method fromJSON():
Usage:
ColumnStatistic$fromJSON(ColumnStatisticJson)

Method toJSONString():
Usage:
ColumnStatistic$toJSONString()

Method fromJSONString():
Usage:
ColumnStatistic$fromJSONString(ColumnStatisticJson)

Method clone(): The objects of this class are cloneable with this method.
Usage:
ColumnStatistic$clone(deep = FALSE)
Arguments:
deep Whether to make a deep clone.

ColumnStatisticsApi ColumnStatistics operations

Description
factset.analyticsapi.engines.ColumnStatistics

Format
An R6Class generator object

Methods

GetPAColumnStatistics Get PA column statistics This endpoint lists all the column statistics that can be applied to a PA column.

@returnType named list( ColumnStatistic )

- On encountering errors, an error of subclass ApiException will be thrown.
- status code : 200 | Expected response, returns a list of PA column statistics
- return type : map(ColumnStatistic)
- response headers :

<table>
<thead>
<tr>
<th>X-RateLimit-Remaining</th>
<th>Number of requests left for the time window.</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-FactSet-Api-Request-Key</td>
<td>Key to uniquely identify an Analytics API request. Only available after successful authentication.</td>
</tr>
<tr>
<td>X-RateLimit-Reset</td>
<td>Number of seconds remaining till rate limit resets.</td>
</tr>
<tr>
<td>X-RateLimit-Limit</td>
<td>Number of allowed requests for the time window.</td>
</tr>
<tr>
<td>X-DataDirect-Request-Key</td>
<td>FactSet's request key header.</td>
</tr>
</tbody>
</table>
• status code : 401 | Missing or invalid authentication.
  • response headers :
    X-RateLimit-Remaining Number of requests left for the time window.
    X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
    X-RateLimit-Reset Number of seconds remaining till rate limit resets.
    X-RateLimit-Limit Number of allowed requests for the time window.
    X-DataDirect-Request-Key FactSet's request key header.

• status code : 403 | User is forbidden with current credentials
  • response headers :
    X-RateLimit-Remaining Number of requests left for the time window.
    X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
    X-RateLimit-Reset Number of seconds remaining till rate limit resets.
    X-RateLimit-Limit Number of allowed requests for the time window.
    X-DataDirect-Request-Key FactSet's request key header.

• status code : 406 | Unsupported Accept header. Header needs to be set to application/json.
  • response headers :
    X-RateLimit-Remaining Number of requests left for the time window.
    X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
    X-RateLimit-Reset Number of seconds remaining till rate limit resets.
    X-RateLimit-Limit Number of allowed requests for the time window.
    X-DataDirect-Request-Key FactSet's request key header.

• status code : 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting
  • response headers :
    X-RateLimit-Remaining Number of requests left for the time window.
    X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
    X-RateLimit-Reset Number of seconds remaining till rate limit resets.
    X-RateLimit-Limit Number of allowed requests for the time window.
    X-DataDirect-Request-Key FactSet's request key header.

• status code : 503 | Request timed out. Retry the request in sometime.
  • response headers :
    X-RateLimit-Remaining Number of requests left for the time window.
    X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
    X-RateLimit-Reset Number of seconds remaining till rate limit resets.
    X-RateLimit-Limit Number of allowed requests for the time window.
    X-DataDirect-Request-Key FactSet's request key header.
X-DataDirect-Request-Key  FactSet’s request key header.

**Public fields**

apiClient  Handles the client-server communication.

**Methods**

**Public methods:**

- `ColumnStatisticsApi$new()`  
- `ColumnStatisticsApi$GetPAColumnStatistics()`  
- `ColumnStatisticsApi$GetPAColumnStatisticsWithHttpInfo()`  
- `ColumnStatisticsApi$clone()`  

**Method** `new()`:

*Usage:*

`ColumnStatisticsApi$new(apiClient)`  

**Method** `GetPAColumnStatistics()`:

*Usage:*

`ColumnStatisticsApi$GetPAColumnStatistics(...)`  

**Method** `GetPAColumnStatisticsWithHttpInfo()`:

*Usage:*

`ColumnStatisticsApi$GetPAColumnStatisticsWithHttpInfo(...)`  

**Method** `clone()`: The objects of this class are cloneable with this method.

*Usage:*

`ColumnStatisticsApi$clone(deep = FALSE)`  

*Arguments:*

- `deep`  Whether to make a deep clone.

**Examples**

```r
## Not run:  
############################ GetPAColumnStatistics ########################

library(factset.analyticsapi.engines)

# Get PA column statistics  
api.instance <- ColumnStatisticsApi$new()

# Configure HTTP basic authorization: Basic  
# provide your username in the user-serial format  
api.instance$apiClient$username <- '<user-serial>';  
# provide your api key generated using the developer portal  
api.instance$apiClient$password <- '<api_key>';  
```
result <- tryCatch(
  api.instance$GetPAColumnStatistics(),
  ApiException = function(ex) ex
)
# In case of error, print the error object
if(!is.null(result$ApiException)) {
  cat(result$ApiException$toString())
} else {
  # deserialized response object
  response.object <- result$content
  # response headers
  response.headers <- result$response$headers
  # response status code
  response.status.code <- result$response$status_code
}

## End(Not run)

#### Description

ColumnSummary Class

#### Format

An R6Class generator object

#### Public fields

- name character [optional]
- directory character [optional]
- category character [optional]

#### Methods

Public methods:

- ColumnSummary$new()
- ColumnSummary$toJSON()
- ColumnSummary$fromJSON()
- ColumnSummary$toJSONString()
- ColumnSummary$fromJSONString()
- ColumnSummary$clone()
**Method new():**

*Usage:*

```
ColumnSummary$new(name = NULL, directory = NULL, category = NULL, ...)
```

**Method toJSON():**

*Usage:*

```
ColumnSummary$toJSON()
```

**Method fromJSON():**

*Usage:*

```
ColumnSummary$fromJSON(ColumnSummaryJson)
```

**Method toJSONString():**

*Usage:*

```
ColumnSummary$toJSONString()
```

**Method fromJSONString():**

*Usage:*

```
ColumnSummary$fromJSONString(ColumnSummaryJson)
```

**Method clone():** The objects of this class are cloneable with this method.

*Usage:*

```
ColumnSummary$clone(deep = FALSE)
```

*Arguments:*

- **deep** Whether to make a deep clone.

---

**Description**

ComponentAccount Class

**Format**

An R6Class generator object

**Public fields**

- **id** character [optional]
- **name** character [optional]
- **holdingsmode** character [optional]
Methods

Public methods:

• `ComponentAccount$new()`
• `ComponentAccount$toJSON()`
• `ComponentAccount$fromJSON()`
• `ComponentAccount$toJSONString()`
• `ComponentAccount$fromJSONString()`
• `ComponentAccount$clone()`

Method `new()`:

Usage:
```
ComponentAccount$new(id = NULL, name = NULL, holdingsmode = NULL, ...)
```

Method `toJSON()`:

Usage:
```
ComponentAccount$toJSON()
```

Method `fromJSON()`:

Usage:
```
ComponentAccount$fromJSON(ComponentAccountJson)
```

Method `toJSONString()`:

Usage:
```
ComponentAccount$toJSONString()
```

Method `fromJSONString()`:

Usage:
```
ComponentAccount$fromJSONString(ComponentAccountJson)
```

Method `clone()`: The objects of this class are cloneable with this method.

Usage:
```
ComponentAccount$clone(deep = FALSE)
```

Arguments:
depth Whether to make a deep clone.
ComponentBenchmark

Description

ComponentBenchmark Class

Format

An R6Class generator object

Public fields

id character [optional]
name character [optional]
holdingsmode character [optional]

Methods

Public methods:

- ComponentBenchmark$new()
- ComponentBenchmark$toJSON()
- ComponentBenchmark$fromJSON()
- ComponentBenchmark$toJSONString()
- ComponentBenchmark$fromJSONString()
- ComponentBenchmark$clone()

Method new():

Usage:
ComponentBenchmark$new(id = NULL, name = NULL, holdingsmode = NULL, ...)

Method toJSON():

Usage:
ComponentBenchmark$toJSON()

Method fromJSON():

Usage:
ComponentBenchmark$fromJSON(ComponentBenchmarkJson)

Method toJSONString():

Usage:
ComponentBenchmark$toJSONString()

Method fromJSONString():
Usage:
ComponentBenchmark$fromJSONString(ComponentBenchmarkJson)

**Method** `clone()`: The objects of this class are cloneable with this method.

**Usage:**
ComponentBenchmark$clone(deep = FALSE)

**Arguments:**
- deep  Whether to make a deep clone.

---

**ComponentsApi**

**Components operations**

---

**Description**

factset.analyticsapi.engines.Components

**Format**

An R6Class generator object

**Methods**

**GetPACOMPONENTBYID**  
*Get PA component by id*  
This endpoint returns the default settings of a PA component.

- **@param**  id character
- **@returnType**  PAComponent

- On encountering errors, an error of subclass ApiException will be thrown.
- **status code : 200 | Expected response, returns the default settings of a PA component.**
- **return type : PAComponent**
- **response headers :**
  - X-RateLimit-Remaining  Number of requests left for the time window.
  - X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit  Number of allowed requests for the time window.
  - X-DataDirect-Request-Key  FactSet#$39;s request key header.

- **status code : 400 | Invalid PA component id.**
- **response headers :**
  - X-RateLimit-Remaining  Number of requests left for the time window.
  - X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-RateLimit-Limit</td>
<td>Number of allowed requests for the time window.</td>
</tr>
<tr>
<td>X-DataDirect-Request-Key</td>
<td>FactSet's request key header.</td>
</tr>
</tbody>
</table>

- status code: 401 | Missing or invalid authentication.                                        |
- response headers:

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-RateLimit-Remaining</td>
<td>Number of requests left for the time window.</td>
</tr>
<tr>
<td>X-FactSet-Api-Request-Key</td>
<td>Key to uniquely identify an Analytics API request. Only available after successful authentication.</td>
</tr>
<tr>
<td>X-RateLimit-Reset</td>
<td>Number of seconds remaining till rate limit resets.</td>
</tr>
<tr>
<td>X-RateLimit-Limit</td>
<td>Number of allowed requests for the time window.</td>
</tr>
<tr>
<td>X-DataDirect-Request-Key</td>
<td>FactSet's request key header.</td>
</tr>
</tbody>
</table>

- status code: 403 | User is forbidden with current credentials                                 |
- response headers:

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-RateLimit-Remaining</td>
<td>Number of requests left for the time window.</td>
</tr>
<tr>
<td>X-FactSet-Api-Request-Key</td>
<td>Key to uniquely identify an Analytics API request. Only available after successful authentication.</td>
</tr>
<tr>
<td>X-RateLimit-Reset</td>
<td>Number of seconds remaining till rate limit resets.</td>
</tr>
<tr>
<td>X-RateLimit-Limit</td>
<td>Number of allowed requests for the time window.</td>
</tr>
<tr>
<td>X-DataDirect-Request-Key</td>
<td>FactSet's request key header.</td>
</tr>
</tbody>
</table>

- status code: 404 | Component not found.                                                        |
- response headers:

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-RateLimit-Remaining</td>
<td>Number of requests left for the time window.</td>
</tr>
<tr>
<td>X-FactSet-Api-Request-Key</td>
<td>Key to uniquely identify an Analytics API request. Only available after successful authentication.</td>
</tr>
<tr>
<td>X-RateLimit-Reset</td>
<td>Number of seconds remaining till rate limit resets.</td>
</tr>
<tr>
<td>X-RateLimit-Limit</td>
<td>Number of allowed requests for the time window.</td>
</tr>
<tr>
<td>X-DataDirect-Request-Key</td>
<td>FactSet's request key header.</td>
</tr>
</tbody>
</table>

- status code: 406 | Unsupported Accept header. Header needs to be set to application/json.      |
- response headers:

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-RateLimit-Remaining</td>
<td>Number of requests left for the time window.</td>
</tr>
<tr>
<td>X-FactSet-Api-Request-Key</td>
<td>Key to uniquely identify an Analytics API request. Only available after successful authentication.</td>
</tr>
<tr>
<td>X-RateLimit-Reset</td>
<td>Number of seconds remaining till rate limit resets.</td>
</tr>
<tr>
<td>X-RateLimit-Limit</td>
<td>Number of allowed requests for the time window.</td>
</tr>
<tr>
<td>X-DataDirect-Request-Key</td>
<td>FactSet's request key header.</td>
</tr>
</tbody>
</table>

- status code: 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting. |
- response headers:
ComponentsApi

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

- status code: 503 | Request timed out. Retry the request in sometime.
- response headers:

GetPAComponents  Get PA components  This endpoint returns the list of PA components in a given PA document.

@ param  document character

@ returnType  named list( ComponentSummary )

- On encountering errors, an error of subclass ApiException will be thrown.
- status code: 200 | Expected response, returns a list of PA components.
- return type: map(ComponentSummary)
- response headers:

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

- status code: 400 | Invalid query parameter provided or Invalid PA document name.
- response headers:

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

- status code: 401 | Missing or invalid authentication.
- response headers:

X-RateLimit-Remaining  Number of requests left for the time window.
- status code: 403 | User is forbidden with current credentials
- response headers:

- status code: 404 | Document not found.
- response headers:

- status code: 406 | Unsupported Accept header. Header needs to be set to application/json.
- response headers:

- status code: 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting.
- response headers:

- status code: 503 | Request timed out. Retry the request in sometime.
GetSPARComponents  

Get SPAR components  This endpoint returns the list of SPAR components in a given SPAR document.

@param  document character
@returnType  named list( ComponentSummary )

On encountering errors, an error of subclass ApiException will be thrown.

status code : 200 | Expected response, returns a list of SPAR components.
return type : map(ComponentSummary)
response headers :

status code : 400 | Invalid query parameter provided or Invalid SPAR document name.
response headers :

status code : 401 | Missing or invalid authentication.
response headers :

status code : 403 | User is forbidden with current credentials
ComponentsApi

• response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet’s request key header.

• status code: 404 | SPAR Document not found.
• response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet’s request key header.

• status code: 406 | Unsupported Accept header. Header needs to be set to application/json.
• response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet’s request key header.

• status code: 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting.
• response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet’s request key header.

• status code: 503 | Request timed out. Retry the request in sometime.
• response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet’s request key header.
GetVaultComponentById  Get Vault component by id  This endpoint returns the default settings of a Vault component.

@param  id character  
@returnType  VaultComponent

- On encountering errors, an error of subclass ApiException will be thrown.
- status code : 200 | Expected response, returns the default settings of a Vault component.
- return type : VaultComponent
- response headers :
  X-RateLimit-Remaining  Number of requests left for the time window.
  X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit  Number of allowed requests for the time window.
  X-DataDirect-Request-Key  FactSet's request key header.

- status code : 400 | Invalid Vault component id.
- response headers :
  X-RateLimit-Remaining  Number of requests left for the time window.
  X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit  Number of allowed requests for the time window.
  X-DataDirect-Request-Key  FactSet's request key header.

- status code : 401 | Missing or invalid authentication.
- response headers :
  X-RateLimit-Remaining  Number of requests left for the time window.
  X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit  Number of allowed requests for the time window.
  X-DataDirect-Request-Key  FactSet's request key header.

- status code : 403 | User is forbidden with current credentials
- response headers :
  X-RateLimit-Remaining  Number of requests left for the time window.
  X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit  Number of allowed requests for the time window.
  X-DataDirect-Request-Key  FactSet's request key header.
ComponentsApi

- status code: 404 | Vault Component not found.
- response headers:

  X-RateLimit-Remaining  
  X-FactSet-Api-Request-Key  
  X-RateLimit-Reset  
  X-RateLimit-Limit  
  X-DataDirect-Request-Key

- status code: 406 | Unsupported Accept header. Header needs to be set to application/json.
- response headers:

  X-RateLimit-Remaining  
  X-FactSet-Api-Request-Key  
  X-RateLimit-Reset  
  X-RateLimit-Limit  
  X-DataDirect-Request-Key

- status code: 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting.
- response headers:

  X-RateLimit-Remaining  
  X-FactSet-Api-Request-Key  
  X-RateLimit-Reset  
  X-RateLimit-Limit  
  X-DataDirect-Request-Key

- status code: 503 | Request timed out. Retry the request in sometime.
- response headers:

  X-RateLimit-Remaining  
  X-FactSet-Api-Request-Key  
  X-RateLimit-Reset  
  X-RateLimit-Limit  
  X-DataDirect-Request-Key

GetVaultComponents  Get Vault components  This endpoint returns the list of Vault components in a given Vault document.

  @param  
  @returnType  named list( ComponentSummary )

- On encountering errors, an error of subclass ApiException will be thrown.
- status code: 200 | Expected response, returns a list of Vault components.
ComponentsApi

- return type: map(ComponentSummary)
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet_Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet's request key header.

- status code: 400 | Invalid query parameter provided or invalid Vault document name.
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet_Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet's request key header.

- status code: 401 | Missing or invalid authentication.
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet_Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet's request key header.

- status code: 403 | User is forbidden with current credentials
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet_Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet's request key header.

- status code: 404 | Vault Document not found.
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet_Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet's request key header.
ComponentsApi

- status code: 406 | Unsupported Accept header. Header needs to be set to application/json.
- response headers:

  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet’s request key header.

- status code: 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting.
- response headers:

  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet’s request key header.

- status code: 503 | Request timed out. Retry the request in sometime.
- response headers:

  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet’s request key header.

Public fields

  apiClient Handles the client-server communication.

Methods

Public methods:

- ComponentsApi$new()
- ComponentsApi$GetPAComponentById()
- ComponentsApi$GetPAComponentByIdWithHttpInfo()
- ComponentsApi$GetPAComponents()
- ComponentsApi$GetPAComponentsWithHttpInfo()
- ComponentsApi$GetSPARComponents()
- ComponentsApi$GetSPARComponentsWithHttpInfo()
ComponentsApi

• ComponentsApi$GetVaultComponentById()
• ComponentsApi$GetVaultComponentByIdWithHttpInfo()
• ComponentsApi$GetVaultComponents()
• ComponentsApi$GetVaultComponentsWithHttpInfo()
• ComponentsApi$clone()

Method new():
   Usage:
   ComponentsApi$new(apiClient)

Method GetPAComponentById():
   Usage:
   ComponentsApi$GetPAComponentById(id, ...)

Method GetPAComponentByIdWithHttpInfo():
   Usage:
   ComponentsApi$GetPAComponentByIdWithHttpInfo(id, ...)

Method GetPAComponents():
   Usage:
   ComponentsApi$GetPAComponents(document, ...)

Method GetPAComponentsWithHttpInfo():
   Usage:
   ComponentsApi$GetPAComponentsWithHttpInfo(document, ...)

Method GetSPARComponents():
   Usage:
   ComponentsApi$GetSPARComponents(document, ...)

Method GetSPARComponentsWithHttpInfo():
   Usage:
   ComponentsApi$GetSPARComponentsWithHttpInfo(document, ...)

Method GetVaultComponentById():
   Usage:
   ComponentsApi$GetVaultComponentById(id, ...)

Method GetVaultComponentByIdWithHttpInfo():
   Usage:
   ComponentsApi$GetVaultComponentByIdWithHttpInfo(id, ...)

Method GetVaultComponents():
   Usage:
   ComponentsApi$GetVaultComponents(document, ...)
**Method** GetVaultComponentsWithHttpInfo():

*Usage:*

ComponentsApi$GetVaultComponentsWithHttpInfo(document, ...)

**Method** clone(): The objects of this class are cloneable with this method.

*Usage:*

ComponentsApi$clone(deep = FALSE)

*Arguments:*

- deep: Whether to make a deep clone.

**Examples**

### Not run:

```
# GetPAComponentById
library(factset.analyticsapi.engines)
var.id <- 'id_example' # character | Unique identifier for a PA component

api.instance <- ComponentsApi$new()

api.instance$apiClient$username <- <user-serial>
api.instance$apiClient$password <- <api_key>;

result <- tryCatch(
    api.instance$GetPAComponentById(var.id),
    ApiException = function(ex) ex
)

if(!is.null(result$ApiException)) {
    cat(result$ApiException$toString())
} else {
    # deserialized response object
    response.object <- result$content
    # response headers
    response.headers <- result$response$headers
    # response status code
    response.status.code <- result$response$status_code
}
```

### GetPAComponents

```
library(factset.analyticsapi.engines)
var.document <- 'document_example' # character | Document Name

api.instance <- ComponentsApi$new()
```
# Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>';  
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>';  
result <- tryCatch(
        api.instance$GetPAComponents(var.document), ApiException = function(ex) ex
    )
# In case of error, print the error object
if(!is.null(result$ApiException)) {
    cat(result$ApiException$toString())
} else {
# deserialized response object
response.object <- result$content
# response headers
response.headers <- result$response$headers
# response status code
response.status.code <- result$response$status_code
}

################################################## GetSPARComponents ##################################################

library(factset.analyticsapi.engines)
var.document <- 'document_example' # character | Document Name

# Get SPAR components
api.instance <- ComponentsApi$new()

# Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>';  
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>';  
result <- tryCatch(
        api.instance$GetSPARComponents(var.document), ApiException = function(ex) ex
    )
# In case of error, print the error object
if(!is.null(result$ApiException)) {
    cat(result$ApiException$toString())
} else {
# deserialized response object
response.object <- result$content
# response headers
response.headers <- result$response$headers
# response status code
response.status.code <- result$response$status_code
}
GetVaultComponentById

```r
library(factset.analyticsapi.engines)
var.id <- 'id_example' # character | Unique identifier for a vault component

# Get Vault component by id
api.instance <- ComponentsApi$new()

# Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>';
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>';

result <- tryCatch(
  api.instance$GetVaultComponentById(var.id),
  ApiException = function(ex) ex
)

# In case of error, print the error object
if(!is.null(result$ApiException)) {
  cat(result$ApiException$toString())
} else {
  # deserialized response object
  response.object <- result$content
  # response headers
  response.headers <- result$response$headers
  # response status code
  response.status.code <- result$response$status_code
}
```

GetVaultComponents

```r
library(factset.analyticsapi.engines)
var.document <- 'document_example' # character | Document Name

# Get Vault components
api.instance <- ComponentsApi$new()

# Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>';
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>';

result <- tryCatch(
  api.instance$GetVaultComponents(var.document),
  ApiException = function(ex) ex
)

# In case of error, print the error object
if(!is.null(result$ApiException)) {
```
cat(result$ApiException$toString())
} else {
  # deserialized response object
  response.object <- result$content
  # response headers
  response.headers <- result$response$headers
  # response status code
  response.status.code <- result$response$status_code
}

## End(Not run)

---

ComponentSummary  ComponentSummary

### Description

ComponentSummary Class

### Format

An R6Class generator object

### Public fields

- **name** character [optional]
- **category** character [optional]

### Methods

**Public methods:**

- ComponentSummary$new()
- ComponentSummary$toJSON()
- ComponentSummary$fromJSON()
- ComponentSummary$toJSONString()
- ComponentSummary$fromJsonString()
- ComponentSummary$clone()

**Method** `new()`:

*Usage:*

ComponentSummary$new(name = NULL, category = NULL, ...)

**Method** `toJSON()`:

*Usage:*


ComponentSummary$toJSON()

**Method** fromJSON():

*Usage:*
ComponentSummary$fromJSON(ComponentSummaryJson)

**Method** toJSONString():

*Usage:*
ComponentSummary$toJSONString()

**Method** fromJSONString():

*Usage:*
ComponentSummary$fromJSONString(ComponentSummaryJson)

**Method** clone(): The objects of this class are cloneable with this method.

*Usage:*
ComponentSummary$clone(deep = FALSE)

*Arguments:*

deep Whether to make a deep clone.

---

**ConfigurationAccount**

**ConfigurationAccount**

---

**Description**

ConfigurationAccount Class

**Format**

An R6Class generator object

**Public fields**

- benchmarkCode character [optional]
- benchmarkName character [optional]
- maxEndDate character [optional]
- minStartDate character [optional]
- lockingDate character [optional]
- name character [optional]
Methods

Public methods:
• ConfigurationAccount$new()
• ConfigurationAccount$toJSON()
• ConfigurationAccount$fromJSON()
• ConfigurationAccount$toJSONString()
• ConfigurationAccount$fromJSONString()
• ConfigurationAccount$clone()

Method new():
Usage:
ConfigurationAccount$new(
  benchmarkCode = NULL,
  benchmarkName = NULL,
  maxEndDate = NULL,
  minStartDate = NULL,
  lockingDate = NULL,
  name = NULL,
  ...
)

Method toJSON():
Usage:
ConfigurationAccount$toJSON()

Method fromJSON():
Usage:
ConfigurationAccount$fromJSON(ConfigurationAccountJson)

Method toJSONString():
Usage:
ConfigurationAccount$toJSONString()

Method fromJSONString():
Usage:
ConfigurationAccount$fromJSONString(ConfigurationAccountJson)

Method clone(): The objects of this class are cloneable with this method.
Usage:
ConfigurationAccount$clone(deep = FALSE)
Arguments:
deep Whether to make a deep clone.
ConfigurationsApi

Configurations operations

Description

factset.analyticsapi.engines.Configurations

Format

An R6Class generator object

Methods

GetVaultConfigurationById  Get Vault configuration by id  This endpoint returns details for a Vault configuration as well as a list of accounts it is used in.

@ param id character

• @ returnType VaultConfiguration

• On encountering errors, an error of subclass ApiException will be thrown.
• status code : 200  Expected response, returns details for a Vault configuration.
• return type : VaultConfiguration
• response headers :

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

• status code : 400  Invalid vault configuration code.
• response headers :

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

• status code : 401  Missing or invalid authentication.
• response headers :

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
## ConfigurationsApi

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-RateLimit-Limit</td>
<td>Number of allowed requests for the time window.</td>
</tr>
<tr>
<td>X-DataDirect-Request-Key</td>
<td>FactSet's request key header.</td>
</tr>
</tbody>
</table>

- status code: 403 | User is forbidden with current credentials
- response headers:

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-RateLimit-Remaining</td>
<td>Number of requests left for the time window.</td>
</tr>
<tr>
<td>X-FactSet-Api-Request-Key</td>
<td>Key to uniquely identify an Analytics API request. Only available after successful authentication.</td>
</tr>
<tr>
<td>X-RateLimit-Reset</td>
<td>Number of seconds remaining till rate limit resets.</td>
</tr>
<tr>
<td>X-RateLimit-Limit</td>
<td>Number of allowed requests for the time window.</td>
</tr>
<tr>
<td>X-DataDirect-Request-Key</td>
<td>FactSet's request key header.</td>
</tr>
</tbody>
</table>

- status code: 404 | Vault configuration id not found.
- response headers:

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-RateLimit-Remaining</td>
<td>Number of requests left for the time window.</td>
</tr>
<tr>
<td>X-FactSet-Api-Request-Key</td>
<td>Key to uniquely identify an Analytics API request. Only available after successful authentication.</td>
</tr>
<tr>
<td>X-RateLimit-Reset</td>
<td>Number of seconds remaining till rate limit resets.</td>
</tr>
<tr>
<td>X-RateLimit-Limit</td>
<td>Number of allowed requests for the time window.</td>
</tr>
<tr>
<td>X-DataDirect-Request-Key</td>
<td>FactSet's request key header.</td>
</tr>
</tbody>
</table>

- status code: 406 | Unsupported Accept header. Header needs to be set to application/json.
- response headers:

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-RateLimit-Remaining</td>
<td>Number of requests left for the time window.</td>
</tr>
<tr>
<td>X-FactSet-Api-Request-Key</td>
<td>Key to uniquely identify an Analytics API request. Only available after successful authentication.</td>
</tr>
<tr>
<td>X-RateLimit-Reset</td>
<td>Number of seconds remaining till rate limit resets.</td>
</tr>
<tr>
<td>X-RateLimit-Limit</td>
<td>Number of allowed requests for the time window.</td>
</tr>
<tr>
<td>X-DataDirect-Request-Key</td>
<td>FactSet's request key header.</td>
</tr>
</tbody>
</table>

- status code: 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting.
- response headers:

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-RateLimit-Remaining</td>
<td>Number of requests left for the time window.</td>
</tr>
<tr>
<td>X-FactSet-Api-Request-Key</td>
<td>Key to uniquely identify an Analytics API request. Only available after successful authentication.</td>
</tr>
<tr>
<td>X-RateLimit-Reset</td>
<td>Number of seconds remaining till rate limit resets.</td>
</tr>
<tr>
<td>X-RateLimit-Limit</td>
<td>Number of allowed requests for the time window.</td>
</tr>
<tr>
<td>X-DataDirect-Request-Key</td>
<td>FactSet's request key header.</td>
</tr>
</tbody>
</table>

- status code: 503 | Request timed out. Retry the request in sometime.
- response headers:
GetVaultConfigurations  Get Vault configurations  This endpoint returns all the Vault configurations saved in the provided account.

@param account character
@returnType named list( VaultConfigurationSummary )

- On encountering errors, an error of subclass ApiException will be thrown.
- status code : 200 | Expected response, returns a dictionary of Vault configurations.
- return type : map(VaultConfigurationSummary)
- response headers :

  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet's request key header.

- status code : 400 | Invalid account.
- response headers :

  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet's request key header.

- status code : 401 | Missing or invalid authentication.
- response headers :

  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet's request key header.

- status code : 403 | User is forbidden with current credentials
- response headers :

  X-RateLimit-Remaining Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset       Number of seconds remaining till rate limit resets.
X-RateLimit-Limit       Number of allowed requests for the time window.
X-DataDirect-Request-Key FactSet's request key header.

• status code : 404 | Account not found.
• response headers :
  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet's request key header.

• status code : 406 | Unsupported Accept header. Header needs to be set to application/json.
• response headers :
  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet's request key header.

• status code : 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting.
• response headers :
  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet's request key header.

• status code : 503 | Request timed out. Retry the request in sometime.
• response headers :
  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet's request key header.
Public fields

apiClient  Handles the client-server communication.

Methods

Public methods:

- ConfigurationsApi$new()
- ConfigurationsApi$GetVaultConfigurationById()
- ConfigurationsApi$GetVaultConfigurationByIdWithHttpInfo()
- ConfigurationsApi$GetVaultConfigurations()
- ConfigurationsApi$GetVaultConfigurationsWithHttpInfo()
- ConfigurationsApi$clone()

Method new():

Usage:
ConfigurationsApi$new(apiClient)

Method GetVaultConfigurationById():

Usage:
ConfigurationsApi$GetVaultConfigurationById(id, ...)

Method GetVaultConfigurationByIdWithHttpInfo():

Usage:
ConfigurationsApi$GetVaultConfigurationByIdWithHttpInfo(id, ...)

Method GetVaultConfigurations():

Usage:
ConfigurationsApi$GetVaultConfigurations(account, ...)

Method GetVaultConfigurationsWithHttpInfo():

Usage:
ConfigurationsApi$GetVaultConfigurationsWithHttpInfo(account, ...)

Method clone(): The objects of this class are cloneable with this method.

Usage:
ConfigurationsApi$clone(deep = FALSE)

Arguments:
deep  Whether to make a deep clone.
### GetVaultConfigurationById

```r
test <- example <-
```

```r
library(factset.analyticsapi.engines)
var.id <- 'id_example'  # character | Vault configuration id to get the details of

#Get Vault configuration by id
api.instance <- ConfigurationsApi$new()

#Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>';
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>';

result <- tryCatch(
  api.instance$GetVaultConfigurationById(var.id),
  ApiException = function(ex) ex
)

# In case of error, print the error object
if(!is.null(result$ApiException)) {
  cat(result$ApiException$toString())
} else {
  # deserialized response object
  response.object <- result$content
  # response headers
  response.headers <- result$response$headers
  # response status code
  response.status.code <- result$response$status_code
}
```

### GetVaultConfigurations

```r
test <- example <-
```

```r
library(factset.analyticsapi.engines)
var.account <- 'account_example'  # character | Required account query parameter to
# filter configurations for a specific account

#Get Vault configurations
api.instance <- ConfigurationsApi$new()

#Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>';
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>';

result <- tryCatch(
  api.instance$GetVaultConfigurations(var.account),
  ApiException = function(ex) ex
)
Currencies operations

Description

factset.analyticsapi.engines.Currencies

Format

An R6Class generator object

Methods

**GetPACurrencies**  
*Get PA currencies*  
This endpoint lists all the PA currencies that can be applied to a calculation.

@returnType named list( Currency )

- On encountering errors, an error of subclass ApiException will be thrown.
- status code : 200 | Expected response, returns a list of PA currencies.
- return type : map(Currency)
- response headers :

  - X-RateLimit-Remaining  
    Number of requests left for the time window.
  - X-FactSet_Api-Request-Key  
    Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset  
    Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit  
    Number of allowed requests for the time window.
  - X-DataDirect-Request-Key  
    FactSet's request key header.

- status code : 401 | Missing or invalid authentication.
- response headers :
X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet's request key header.

• status code : 403 | User is forbidden with current credentials
• response headers :

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet's request key header.

• status code : 406 | Unsupported Accept header. Header needs to be set to application/json.
• response headers :

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet's request key header.

• status code : 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting.
• response headers :

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet's request key header.

• status code : 503 | Request timed out. Retry the request in sometime.
• response headers :

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet's request key header.
Public fields

apiClient  Handles the client-server communication.

Methods

Public methods:

• CurrenciesApi$new()
• CurrenciesApi$GetPACurrencies()
• CurrenciesApi$GetPACurrenciesWithHttpInfo()
• CurrenciesApi$clone()

Method new():

Usage:
CurrenciesApi$new(apiClient)

Method GetPACurrencies():

Usage:
CurrenciesApi$GetPACurrencies(...)

Method GetPACurrenciesWithHttpInfo():

Usage:
CurrenciesApi$GetPACurrenciesWithHttpInfo(...)

Method clone(): The objects of this class are cloneable with this method.

Usage:
CurrenciesApi$clone(deep = FALSE)

Arguments:

| deep | Whether to make a deep clone. |

Examples

```r
## Not run:

# Get PA currencies
api.instance <- CurrenciesApi$new()

# Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>'
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>'

result <- tryCatch(
    api.instance$GetPACurrencies(),
```
ApiException = function(ex) ex
)
# In case of error, print the error object
if(!is.null(result$ApiException)) {
  cat(result$ApiException$toString())
} else {
  # deserialized response object
  response.object <- result$content
  # response headers
  response.headers <- result$response$headers
  # response status code
  response.status.code <- result$response$status_code
}

## End(Not run)

---

**Currency**

**Description**

Currency Class

**Format**

An R6Class generator object

**Public fields**

- name character [optional]

**Methods**

**Public methods:**

- Currency$new()
- Currency$toJSON()
- Currency$fromJSON()
- Currency$toJSONString()
- Currency$fromJSONString()
- Currency$clone()

**Method** `new()`:

*Usage:*

Currency$new(name = NULL, ...)

**Method** `toJSON()`:
Usage:
Currency$toJSON()

Method fromJSON():
Usage:
Currency$fromJSON(CurrencyJson)

Method toJSONString():
Usage:
Currency$toJSONString()

Method fromJSONString():
Usage:
Currency$fromJSONString(CurrencyJson)

Method clone(): The objects of this class are cloneable with this method.
Usage:
Currency$clone(deep = FALSE)
Arguments:
deep  Whether to make a deep clone.

DateParametersSummary

Description
DateParametersSummary Class

Format
An R6Class generator object

Public fields
startdate character [optional]
enddate character [optional]
Methods

**Public methods:**

- `DateParametersSummary$new()`
- `DateParametersSummary$toJSON()`
- `DateParametersSummary$fromJSON()`
- `DateParametersSummary$toJSONString()`
- `DateParametersSummary$fromJSONString()`
- `DateParametersSummary$clone()`

**Method** `new()`:

**Usage:**

```
DateParametersSummary$new(startdate = NULL, enddate = NULL, ...)
```

**Method** `toJSON()`:

**Usage:**

```
DateParametersSummary$toJSON()
```

**Method** `fromJSON()`:

**Usage:**

```
DateParametersSummary$fromJSON(DateParametersSummaryJson)
```

**Method** `toJSONString()`:

**Usage:**

```
DateParametersSummary$toJSONString()
```

**Method** `fromJSONString()`:

**Usage:**

```
DateParametersSummary$fromJSONString(DateParametersSummaryJson)
```

**Method** `clone()`: The objects of this class are cloneable with this method.

**Usage:**

```
DateParametersSummary$clone(deep = FALSE)
```

**Arguments:**

depth Whether to make a deep clone.
DatesApi

Dates operations

Description

factset.analyticsapi.engines.Dates

Format

An R6Class generator object

Methods

ConvertPADatesToAbsoluteFormat  
Convert PA dates to absolute format  
This endpoint converts the given start and end dates to yyyymmdd format for a PA calculation.

@param enddate character
@param componentid character
@param account character
@param startdate character
@returnType DateParametersSummary

On encountering errors, an error of subclass ApiException will be thrown.

status code : 200 | Expected response, returns the converted dates in yyyymmdd format.
return type : DateParametersSummary
response headers :

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

status code : 400 | Invalid query parameter or value is provided.
response headers :

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

status code : 401 | Missing or invalid authentication.
response headers :
- status code: 403 | User is forbidden with current credentials
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet's request key header.

- status code: 404 | Document or account not found.
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet's request key header.

- status code: 406 | Unsupported Accept header. Header needs to be set to application/json.
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet's request key header.

- status code: 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting.
- response headers:
DatesApi

- status code: 503 | Request timed out. Retry the request in sometime.
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet’s request key header.

**ConvertVaultDatesToAbsoluteFormat**  
*Convert Vault dates to absolute format*  
This endpoint converts the given start and end dates to yyyyymmdd format for a Vault calculation.

- @param enddate character
- @param componentid character
- @param account character
- @param startdate character
- @returnType DateParametersSummary

- On encountering errors, an error of subclass ApiException will be thrown.
- status code: 200 | Expected response, returns the converted dates in yyyyymmdd format.
- return type: DateParametersSummary
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet’s request key header.

- status code: 400 | Invalid query parameter or value is provided.
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
  - X-DataDirect-Request-Key: FactSet’s request key header.

- status code: 401 | Missing or invalid authentication.
- response headers:
  - X-RateLimit-Remaining: Number of requests left for the time window.
  - X-FactSet-Api-Request-Key: Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-RateLimit-Reset: Number of seconds remaining till rate limit resets.
  - X-RateLimit-Limit: Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

- status code: 403 | User is forbidden with current credentials
- response headers:
  X-RateLimit-Remaining  Number of requests left for the time window.
  X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit  Number of allowed requests for the time window.
  X-DataDirect-Request-Key  FactSet’s request key header.

- status code: 404 | vault Document or account not found.
- response headers:
  X-RateLimit-Remaining  Number of requests left for the time window.
  X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit  Number of allowed requests for the time window.
  X-DataDirect-Request-Key  FactSet’s request key header.

- status code: 406 | Unsupported Accept header. Header needs to be set to application/json.
- response headers:
  X-RateLimit-Remaining  Number of requests left for the time window.
  X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit  Number of allowed requests for the time window.
  X-DataDirect-Request-Key  FactSet’s request key header.

- status code: 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting.
- response headers:
  X-RateLimit-Remaining  Number of requests left for the time window.
  X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit  Number of allowed requests for the time window.
  X-DataDirect-Request-Key  FactSet’s request key header.

- status code: 503 | Request timed out. Retry the request in sometime.
- response headers:
  X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset Number of seconds remaining till rate limit resets.
X-RateLimit-Limit Number of allowed requests for the time window.
X-DataDirect-Request-Key FactSet's request key header.

Public fields

apiClient Handles the client-server communication.

Methods

Public methods:

- DatesApi$new()
- DatesApi$ConvertPADatesToAbsoluteFormat()
- DatesApi$ConvertPADatesToAbsoluteFormatWithHttpInfo()
- DatesApi$ConvertVaultDatesToAbsoluteFormat()
- DatesApi$ConvertVaultDatesToAbsoluteFormatWithHttpInfo()
- DatesApi$clone()

Method new():

Usage:
DatesApi$new(apiClient)

Method ConvertPADatesToAbsoluteFormat():

Usage:
DatesApi$ConvertPADatesToAbsoluteFormat(
  enddate,
  componentid,
  account,
  startdate = NULL,
  ...
)

Method ConvertPADatesToAbsoluteFormatWithHttpInfo():

Usage:
DatesApi$ConvertPADatesToAbsoluteFormatWithHttpInfo(
  enddate,
  componentid,
  account,
  startdate = NULL,
  ...
)

Method ConvertVaultDatesToAbsoluteFormat():

Usage:
DatesApi

DatesApi$ConvertVaultDatesToAbsoluteFormat(
   enddate,
   componentid,
   account,
   startdate = NULL,
   ...
)

Method ConvertVaultDatesToAbsoluteFormatWithHttpInfo():
Usage:
DatesApi$ConvertVaultDatesToAbsoluteFormatWithHttpInfo(
   enddate,
   componentid,
   account,
   startdate = NULL,
   ...
)

Method clone(): The objects of this class are cloneable with this method.
Usage:
DatesApi$clone(deep = FALSE)
Arguments:
dee p Whether to make a deep clone.

Examples

## Not run:
############################## ConvertPADatesToAbsoluteFormat ##############################

library(factset.analyticsapi.engines)
var.enddate <- 'enddate_example' # character | End Date
var.componentid <- 'componentid_example' # character | Component Id
var.account <- 'account_example' # character | Account
var.startdate <- 'startdate_example' # character | Start Date

#Convert PA dates to absolute format
api.instance <- DatesApi$new()

#Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>,'
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>,'

result <- tryCatch(
   api.instance$ConvertPADatesToAbsoluteFormat(var.enddate, var.componentid,
   var.account, startdate=var.startdate),
   ApiException = function(ex) ex
)

# In case of error, print the error object
if(!is.null(result.ApiException)) {
  cat(result.ApiException$toString())
} else {
  # deserialized response object
  response.object <- result$content
  # response headers
  response.headers <- result$response$headers
  # response status code
  response.status.code <- result$response$status_code
}

library(factset.analyticsapi.engines)
var.enddate <- 'enddate_example' # character | End Date
var.componentid <- 'componentid_example' # character | Vault Component Id
var.account <- 'account_example' # character | Account
var.startdate <- 'startdate_example' # character | Start Date

# Convert Vault dates to absolute format
api.instance <- DatesApi$new()

# Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>';
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>';

result <- tryCatch(
  api.instance$ConvertVaultDatesToAbsoluteFormat(var.enddate, var.componentid,
                                              var.account, startdate=var.startdate),
  ApiException = function(ex) ex
)

# In case of error, print the error object
if(!is.null(result$ApiException)) {
  cat(result$ApiException$toString())
} else {
  # deserialized response object
  response.object <- result$content
  # response headers
  response.headers <- result$response$headers
  # response status code
  response.status.code <- result$response$status_code
}

## End(Not run)
**Description**

DocumentDirectories Class

**Format**

An R6Class generator object

**Public fields**

documents list(character) [optional]
directories list(character) [optional]

**Methods**

**Public methods:**
- DocumentDirectories$new()
- DocumentDirectories$toJSON()
- DocumentDirectories$fromJSON()
- DocumentDirectories$toJSONString()
- DocumentDirectories$fromJSONString()
- DocumentDirectories$clone()

**Method** new():

*Usage:*
DocumentDirectories$new(documents = NULL, directories = NULL, ...)

**Method** toJSON():

*Usage:*
DocumentDirectories$toJSON()

**Method** fromJSON():

*Usage:*
DocumentDirectories$fromJSON(DocumentDirectoriesJson)

**Method** toJSONString():

*Usage:*
DocumentDirectories$toJSONString()

**Method** fromJSONString():

*Usage:*
DocumentDirectories$fromJSONString(DocumentDirectoriesJson)

**Method** clone(): The objects of this class are cloneable with this method.

*Usage:*
DocumentDirectories$clone(deep = FALSE)

*Arguments:*
deep Whether to make a deep clone.
Description

factset.analyticsapi.engines.Documents

Format

An R6Class generator object

Methods

**GetPA3Documents**  *Get PA3 documents and sub-directories in a directory*  This endpoint looks up all PA3 documents and sub-directories in a given directory.

@ param  path character

* @ returnType  DocumentDirectories

* On encountering errors, an error of subclass ApiException will be thrown.
* status code : 200 | Expected response, returns a list of PA3 documents and directories
* return type : DocumentDirectories
* response headers :

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

* status code : 400 | Invalid query parameter or value provided
* response headers :

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

* status code : 401 | Missing or invalid authentication
* response headers :

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

- status code: 403 | User is forbidden with current credentials
- response headers:

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

- status code: 404 | Path not found
- response headers:

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

- status code: 406 | Unsupported Accept header. Header needs to be set to application/json
- response headers:

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

- status code: 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting
- response headers:

X-RateLimit-Remaining  Number of requests left for the time window.
X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
X-RateLimit-Limit  Number of allowed requests for the time window.
X-DataDirect-Request-Key  FactSet’s request key header.

- status code: 503 | Request timed out. Retry the request in some time
- response headers:
**GetSPAR3Documents**  
*Gets SPAR3 documents and sub-directories in a directory*  
This endpoint looks up all SPAR3 documents and sub-directories in a given directory.

@param path character

@returnType DocumentDirectories

- On encountering errors, an error of subclass ApiException will be thrown.
- status code : 200 | Expected response, returns a list of SPAR3 documents and directories
- return type : DocumentDirectories
- response headers :

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-RateLimit-Remaining</td>
<td>Number of requests left for the time window.</td>
</tr>
<tr>
<td>X-FactSet-Api-Request-Key</td>
<td>Key to uniquely identify an Analytics API request. Only available after successful authentication.</td>
</tr>
<tr>
<td>X-RateLimit-Reset</td>
<td>Number of seconds remaining till rate limit resets.</td>
</tr>
<tr>
<td>X-RateLimit-Limit</td>
<td>Number of allowed requests for the time window.</td>
</tr>
<tr>
<td>X-DataDirect-Request-Key</td>
<td>FactSet’s request key header.</td>
</tr>
</tbody>
</table>

- status code : 400 | Invalid query parameter or value provided
- response headers :

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-RateLimit-Remaining</td>
<td>Number of requests left for the time window.</td>
</tr>
<tr>
<td>X-FactSet-Api-Request-Key</td>
<td>Key to uniquely identify an Analytics API request. Only available after successful authentication.</td>
</tr>
<tr>
<td>X-RateLimit-Reset</td>
<td>Number of seconds remaining till rate limit resets.</td>
</tr>
<tr>
<td>X-RateLimit-Limit</td>
<td>Number of allowed requests for the time window.</td>
</tr>
<tr>
<td>X-DataDirect-Request-Key</td>
<td>FactSet’s request key header.</td>
</tr>
</tbody>
</table>

- status code : 401 | Missing or invalid authentication
- response headers :

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-RateLimit-Remaining</td>
<td>Number of requests left for the time window.</td>
</tr>
<tr>
<td>X-FactSet-Api-Request-Key</td>
<td>Key to uniquely identify an Analytics API request. Only available after successful authentication.</td>
</tr>
<tr>
<td>X-RateLimit-Reset</td>
<td>Number of seconds remaining till rate limit resets.</td>
</tr>
<tr>
<td>X-RateLimit-Limit</td>
<td>Number of allowed requests for the time window.</td>
</tr>
<tr>
<td>X-DataDirect-Request-Key</td>
<td>FactSet’s request key header.</td>
</tr>
</tbody>
</table>

- status code : 403 | User is forbidden with current credentials
- response headers :

<table>
<thead>
<tr>
<th>Header</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>X-RateLimit-Remaining</td>
<td>Number of requests left for the time window.</td>
</tr>
</tbody>
</table>
GetVaultDocuments  

Get Vault documents and sub-directories in a directory  

This endpoint
DocumentsApi

looks up all Vault documents and sub-directories in a given directory.

@para**m** path character

• @**r**e**t**urn**T**ype  **D**ocumentD**i**rectories

• On encountering errors, an error of subclass ApiException will be thrown.
• status code : 200 | Expected response, returns a list of Vault documents and directories
• return type : DocumentDirectories
• response headers :

  X-RateLimit-Remaining  Number of requests left for the time window.
  X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit  Number of allowed requests for the time window.
  X-DataDirect-Request-Key  FactSet's request key header.

• status code : 400 | Invalid query parameter or value provided
• response headers :

  X-RateLimit-Remaining  Number of requests left for the time window.
  X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit  Number of allowed requests for the time window.
  X-DataDirect-Request-Key  FactSet's request key header.

• status code : 401 | Missing or invalid authentication
• response headers :

  X-RateLimit-Remaining  Number of requests left for the time window.
  X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit  Number of allowed requests for the time window.
  X-DataDirect-Request-Key  FactSet's request key header.

• status code : 403 | User is forbidden with current credentials
• response headers :

  X-RateLimit-Remaining  Number of requests left for the time window.
  X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit  Number of allowed requests for the time window.
  X-DataDirect-Request-Key  FactSet's request key header.

• status code : 404 | Path not found
Public fields

apiClient Handles the client-server communication.

Methods

Public methods:

- DocumentsApi$new()
• DocumentsApi$GetPA3Documents()
• DocumentsApi$GetPA3DocumentsWithHttpInfo()
• DocumentsApi$GetSPAR3Documents()
• DocumentsApi$GetSPAR3DocumentsWithHttpInfo()
• DocumentsApi$GetVaultDocuments()
• DocumentsApi$GetVaultDocumentsWithHttpInfo()
• DocumentsApi$clone()

**Method** new():
Usage:
DocumentsApi$new(apiClient)

**Method** GetPA3Documents():
Usage:
DocumentsApi$GetPA3Documents(path, ...)

**Method** GetPA3DocumentsWithHttpInfo():
Usage:
DocumentsApi$GetPA3DocumentsWithHttpInfo(path, ...)

**Method** GetSPAR3Documents():
Usage:
DocumentsApi$GetSPAR3Documents(path, ...)

**Method** GetSPAR3DocumentsWithHttpInfo():
Usage:
DocumentsApi$GetSPAR3DocumentsWithHttpInfo(path, ...)

**Method** GetVaultDocuments():
Usage:
DocumentsApi$GetVaultDocuments(path, ...)

**Method** GetVaultDocumentsWithHttpInfo():
Usage:
DocumentsApi$GetVaultDocumentsWithHttpInfo(path, ...)

**Method** clone(): The objects of this class are cloneable with this method.
Usage:
DocumentsApi$clone(deep = FALSE)

Arguments:
deep  Whether to make a deep clone.
Examples

### Not run:

```
#################################################################
# GetPA3Documents #################################################################

library(factset.analyticsapi.engines)
var.path <- 'path_example' # character | The directory to get the documents and sub-directories in

#Get PA3 documents and sub-directories in a directory
api.instance <- DocumentsApi$new()

#Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>';
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>';

result <- tryCatch(
  api.instance$GetPA3Documents(var.path),
  ApiException = function(ex) ex
)
# In case of error, print the error object
if(!is.null(result$ApiException)) {
  cat(result$ApiException$toString())
} else {
  # deserialized response object
  response.object <- result$content
  # response headers
  response.headers <- result$response$headers
  # response status code
  response.status.code <- result$response$status_code
}

#################################################################
# GetSPAR3Documents #################################################################

library(factset.analyticsapi.engines)
var.path <- 'path_example' # character | The directory to get the documents in

#Gets SPAR3 documents and sub-directories in a directory
api.instance <- DocumentsApi$new()

#Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>';
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>';

result <- tryCatch(
  api.instance$GetSPAR3Documents(var.path),
  ApiException = function(ex) ex
)
# In case of error, print the error object
if(!is.null(result$ApiException)) {
    cat(result$ApiException$toString())
} else {
# deserialized response object
response.object <- result$content
# response headers
response.headers <- result$response$headers
# response status code
response.status.code <- result$response$status_code
}

#################### GetVaultDocuments ####################

library(factset.analyticsapi.engines)

var.path <- 'path_example' # character | The directory to get the documents in

# Get Vault documents and sub-directories in a directory
api.instance <- DocumentsApi$new()

# Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>';
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>';

result <- tryCatch(
    api.instance$GetVaultDocuments(var.path),
    ApiException = function (ex) ex
)

# In case of error, print the error object
if(!is.null(result$ApiException)) {
    cat(result$ApiException$toString())
} else {
# deserialized response object
response.object <- result$content
# response headers
response.headers <- result$response$headers
# response status code
response.status.code <- result$response$status_code
}

## End(Not run)

---

**FrequenciesApi**  
**Frequencies operations**

**Description**

factset.analyticsapi.engines.Frequencies
Format
An R6Class generator object

Methods
GetPAFrequencies  Get PA frequencies  This endpoint lists all the frequencies that can be applied to a PA calculation.

@returnType named list( Frequency )

• On encountering errors, an error of subclass ApiException will be thrown.
• status code : 200 | Expected response, returns a list of PA frequencies.
• return type : map(Frequency)
• response headers :
  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet_Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet's request key header.

• status code : 401 | Missing or invalid authentication.
• response headers :
  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet_Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet's request key header.

• status code : 403 | User is forbidden with current credentials
• response headers :
  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet_Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet's request key header.

• status code : 406 | Unsupported Accept header. Header needs to be set to application/json.
• response headers :
  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet_Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.

  X-RateLimit-Limit Number of allowed requests for the time window.
GetSPARFrequencies  

*Get SPAR frequencies* This endpoint lists all the frequencies that can be applied to a SPAR calculation.

@returnType  named list( Frequency )

- On encountering errors, an error of subclass ApiException will be thrown.
- status code : 200 | Expected response, returns a list of SPAR frequencies.
- return type : map(Frequency)
- response headers :

  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet's request key header.

- status code : 401 | Missing or invalid authentication.
- response headers :

  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet's request key header.
X-DataDirect-Request-Key  FactSet's request key header.

- status code : 403 | User is forbidden with current credentials
- response headers :

  X-RateLimit-Remaining  Number of requests left for the time window.
  X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit  Number of allowed requests for the time window.
  X-DataDirect-Request-Key  FactSet's request key header.

- status code : 406 | Unsupported Accept header. Header needs to be set to application/json.
- response headers :

  X-RateLimit-Remaining  Number of requests left for the time window.
  X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit  Number of allowed requests for the time window.
  X-DataDirect-Request-Key  FactSet's request key header.

- status code : 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting.
- response headers :

  X-RateLimit-Remaining  Number of requests left for the time window.
  X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit  Number of allowed requests for the time window.
  X-DataDirect-Request-Key  FactSet's request key header.

- status code : 503 | Request timed out. Retry the request in sometime.
- response headers :

  X-RateLimit-Remaining  Number of requests left for the time window.
  X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset  Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit  Number of allowed requests for the time window.
  X-DataDirect-Request-Key  FactSet's request key header.

GetVaultFrequencies  Get Vault frequencies  This endpoint lists all the frequencies that can be applied to a Vault calculation.

@returnType  named list( Frequency )
- On encountering errors, an error of subclass ApiException will be thrown.
- status code : 200 | Expected response, returns a list of Vault frequencies.
- return type : map(Frequency)
- response headers :

  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet’s request key header.

- status code : 401 | Missing or invalid authentication.
- response headers :

  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet’s request key header.

- status code : 403 | User is forbidden with current credentials
- response headers :

  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet’s request key header.

- status code : 406 | Unsupported Accept header. Header needs to be set to application/json.
- response headers :

  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  X-RateLimit-Reset Number of seconds remaining till rate limit resets.
  X-RateLimit-Limit Number of allowed requests for the time window.
  X-DataDirect-Request-Key FactSet’s request key header.

- status code : 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting.
- response headers :

  X-RateLimit-Remaining Number of requests left for the time window.
  X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
FrequenciesApi

X-RateLimit-Reset Number of seconds remaining till rate limit resets.
X-RateLimit-Limit Number of allowed requests for the time window.
X-DataDirect-Request-Key FactSet's request key header.

• status code : 503 | Request timed out. Retry the request in sometime.
• response headers :

X-RateLimit-Remaining Number of requests left for the time window.
X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset Number of seconds remaining till rate limit resets.
X-RateLimit-Limit Number of allowed requests for the time window.
X-DataDirect-Request-Key FactSet's request key header.

Public fields

apiClient Handles the client-server communication.

Methods

Public methods:

• FrequenciesApi$new()
• FrequenciesApi$GetPAFrequencies()
• FrequenciesApi$GetPAFrequenciesWithHttpInfo()
• FrequenciesApi$GetSPARFrequencies()
• FrequenciesApi$GetSPARFrequenciesWithHttpInfo()
• FrequenciesApi$GetVaultFrequencies()
• FrequenciesApi$GetVaultFrequenciesWithHttpInfo()
• FrequenciesApi$clone()

Method new():
Usage:
FrequenciesApi$new(apiClient)

Method GetPAFrequencies():
Usage:
FrequenciesApi$GetPAFrequencies(...) 

Method GetPAFrequenciesWithHttpInfo():
Usage:
FrequenciesApi$GetPAFrequenciesWithHttpInfo(...) 

Method GetSPARFrequencies():
Usage:
FrequenciesApi$GetSPARFrequencies(...)
Method GetSPARFrequenciesWithHttpInfo():
Usage:
FrequenciesApi$GetSPARFrequenciesWithHttpInfo(...)

Method GetVaultFrequencies():
Usage:
FrequenciesApi$GetVaultFrequencies(...)

Method GetVaultFrequenciesWithHttpInfo():
Usage:
FrequenciesApi$GetVaultFrequenciesWithHttpInfo(...)

Method clone(): The objects of this class are cloneable with this method.
Usage:
FrequenciesApi$clone(deep = FALSE)
Arguments:
depth Whether to make a deep clone.

Examples

## Not run:

### GetPAFrequencies

```r
library(factset.analyticsapi.engines)

# Get PA frequencies
api.instance <- FrequenciesApi$new()

# Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>';
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>';

result <- tryCatch(
    api.instance$GetPAFrequencies(),
    ApiException = Function(ex) ex
)

# In case of error, print the error object
if(!is.null(result$ApiException)) {
    cat(result$ApiException$toString())
} else {
# deserialize response object
response.object <- result$content
# response headers
response.headers <- result$response$headers
# response status code
response.status.code <- result$response$status_code
}```
FrequenciesApi

GetSPARFrequencies

# Get SPAR frequencies
api.instance <- FrequenciesApi$new()

# Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>'
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>'

result <- tryCatch(
    api.instance$GetSPARFrequencies(),
    ApiException = function(ex) ex
)

# In case of error, print the error object
if(!is.null(result$ApiException)) {
    cat(result$ApiException$toString())
} else {
    # deserialized response object
    response.object <- result$content
    # response headers
    response.headers <- result$response$headers
    # response status code
    response.status.code <- result$response$status_code
}

GetVaultFrequencies

# Get Vault frequencies
api.instance <- FrequenciesApi$new()

# Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>'
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>'

result <- tryCatch(
    api.instance$GetVaultFrequencies(),
    ApiException = function(ex) ex
)

# In case of error, print the error object
if(!is.null(result$ApiException)) {
    cat(result$ApiException$toString())
} else {

Frequency

Description
Frequency Class

Format
An R6Class generator object

Public fields
name character [optional]

Methods
Public methods:
• Frequency$new()
• Frequency$toJSON()
• Frequency$fromJSON()
• Frequency$toJSONString()
• Frequency$fromJSONString()
• Frequency$clone()

Method new():
Usage:
Frequency$new(name = NULL, ...)

Method toJSON():
Usage:
Frequency$toJSON()

Method fromJSON():
Usage:
Frequency$fromJSON(FrequencyJson)

**Method toJSONString():**

*Usage:*
Frequency$toJSONString()

**Method fromJSONString():**

*Usage:*
Frequency$fromJSONString(FrequencyJson)

**Method clone():** The objects of this class are cloneable with this method.

*Usage:*
Frequency$clone(deep = FALSE)

*Arguments:*
depth Whether to make a deep clone.

---

**Description**

Group Class

**Format**

An R6Class generator object

**Public fields**

- name character [optional]
- directory character [optional]
- category character [optional]

**Methods**

**Public methods:**
- Group$new()
- Group$toJSON()
- Group$fromJSON()
- Group$toJSONString()
- Group$fromJSONString()
- Group$clone()
- Group$clone()

**Method new():**
Usage:
Group$new(name = NULL, directory = NULL, category = NULL, ...)

Method toJSON():
Usage:
Group$toJSON()

Method fromJSON():
Usage:
Group$fromJSON(GroupJson)

Method toJSONString():
Usage:
Group$toJSONString()

Method fromJSONString():
Usage:
Group$fromJSONString(GroupJson)

Method clone(): The objects of this class are cloneable with this method.
Usage:
Group$clone(deep = FALSE)
Arguments:
deep Whether to make a deep clone.

---

GroupsApi

Groups operations

Description

factset.analyticsapi.engines.Groups

Format

An R6Class generator object

Methods

GetPAGroups  Get PA groups  This endpoint lists all the PA groups that can be applied to a PA calculation.

@param returnType named list( Group )

- On encountering errors, an error of subclass ApiException will be thrown.
- status code : 200 | Expected response, returns a list of PA groups
- return type : map(Group)
- response headers :
GroupsApi

X-RateLimit-Remaining Number of requests left for the time window.
X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset Number of seconds remaining till rate limit resets.
X-RateLimit-Limit Number of allowed requests for the time window.
Age Standard HTTP header. Header will specify the age of groupings list cached response.
X-DataDirect-Request-Key FactSet’s request key header.

- status code : 401 | Missing or invalid authentication.
- response headers :

X-RateLimit-Remaining Number of requests left for the time window.
X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset Number of seconds remaining till rate limit resets.
X-RateLimit-Limit Number of allowed requests for the time window.
X-DataDirect-Request-Key FactSet’s request key header.

- status code : 403 | User is forbidden with current credentials
- response headers :

X-RateLimit-Remaining Number of requests left for the time window.
X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset Number of seconds remaining till rate limit resets.
X-RateLimit-Limit Number of allowed requests for the time window.
X-DataDirect-Request-Key FactSet’s request key header.

- status code : 406 | Unsupported Accept header. Header needs to be set to application/json.
- response headers :

X-RateLimit-Remaining Number of requests left for the time window.
X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-RateLimit-Reset Number of seconds remaining till rate limit resets.
X-RateLimit-Limit Number of allowed requests for the time window.
X-DataDirect-Request-Key FactSet’s request key header.

- status code : 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting
- response headers :
• status code : 503 | Request timed out. Retry the request in sometime.
• response headers :

   X-RateLimit-Remaining | Number of requests left for the time window.
   X-FactSet-Api-Request-Key | Key to uniquely identify an Analytics API request. Only available after successful authentication.
   X-RateLimit-Reset | Number of seconds remaining till rate limit resets.
   X-RateLimit-Limit | Number of allowed requests for the time window.
   X-DataDirect-Request-Key | FactSet's request key header.

Public fields

    apiClient | Handles the client-server communication.

Methods

Public methods:
• GroupsApi$new()
• GroupsApi$GetPAGroups()
• GroupsApi$GetPAGroupsWithHttpInfo()
• GroupsApi$clone()

Method new():
    Usage:
    GroupsApi$new(apiClient)

Method GetPAGroups():
    Usage:
    GroupsApi$GetPAGroups(...)

Method GetPAGroupsWithHttpInfo():
    Usage:
    GroupsApi$GetPAGroupsWithHttpInfo(...)

Method clone(): The objects of this class are cloneable with this method.
    Usage:
    GroupsApi$clone(deep = FALSE)

Arguments:
    deep | Whether to make a deep clone.

Examples

## Not run:

# Start of example code
library(factset.analyticsapi.engines)
#Get PA groups
api.instance <- GroupsApi$new()

#Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>';
# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>';

result <- tryCatch(
  api.instance$GetPAGroups(),
  ApiException = function(ex) ex
)
# In case of error, print the error object
if(!is.null(result$ApiException)) {
  cat(result$ApiException$toString())
} else {
  # deserialized response object
  response.object <- result$content
  # response headers
  response.headers <- result$response$headers
  # response status code
  response.status.code <- result$response$status_code
}

## End(Not run)

---

**PACalculationColumn Class**

**Description**

PACalculationColumn Class

**Format**

An R6Class generator object

**Public fields**

- id character [optional]
- statistics list(character) [optional]

**Methods**

- **Public methods:**
  - PACalculationColumn$new()
• PACalculationColumn$toJSON()
• PACalculationColumn$fromJson()
• PACalculationColumn$toJSONString()
• PACalculationColumn$fromJsonString()
• PACalculationColumn$clone()

Method new():
  Usage:
  PACalculationColumn$new(id = NULL, statistics = NULL, ...)

Method toJSON():
  Usage:
  PACalculationColumn$toJSON()

Method fromJSON():
  Usage:
  PACalculationColumn$fromJson(PACalculationColumnJson)

Method toJSONString():
  Usage:
  PACalculationColumn$toJSONString()

Method fromJSONString():
  Usage:
  PACalculationColumn$fromJsonString(PACalculationColumnJson)

Method clone(): The objects of this class are cloneable with this method.
  Usage:
  PACalculationColumn$clone(deep = FALSE)
  Arguments:
  deep  Whether to make a deep clone.

Description
PACalculationGroup Class

Format
An R6Class generator object
PACalculationGroup

Public fields

id character [optional]

Methods

Public methods:

• PACalculationGroup$new()
• PACalculationGroup$toJSON()
• PACalculationGroup$fromJSON()
• PACalculationGroup$toJSONString()
• PACalculationGroup$fromJSONString()
• PACalculationGroup$clone()

Method new():

Usage:
PACalculationGroup$new(id = NULL, ...)

Method toJSON():

Usage:
PACalculationGroup$toJSON()

Method fromJSON():

Usage:
PACalculationGroup$fromJSON(PACalculationGroupJson)

Method toJSONString():

Usage:
PACalculationGroup$toJSONString()

Method fromJSONString():

Usage:
PACalculationGroup$fromJSONString(PACalculationGroupJson)

Method clone(): The objects of this class are cloneable with this method.

Usage:
PACalculationGroup$clone(deep = FALSE)

Arguments:
deep Whether to make a deep clone.
Description

PACalculationParameters Class

Format

An R6Class generator object

Public fields

- componentid character
- accounts list( PAIdentifier ) [optional]
- benchmarks list( PAIdentifier ) [optional]
- dates PADateParameters [optional]
- groups list( PACalculationGroup ) [optional]
- currencyisocode character [optional]
- columns list( PACalculationColumn ) [optional]

Methods

Public methods:

- PACalculationParameters$new()
- PACalculationParameters$toJSON()
- PACalculationParameters$fromJSON()
- PACalculationParameters$toJSONString()
- PACalculationParameters$fromJSONString()
- PACalculationParameters$clone()

Method new():

Usage:

PACalculationParameters$new(  
  componentid,  
  accounts = NULL,  
  benchmarks = NULL,  
  dates = NULL,  
  groups = NULL,  
  currencyisocode = NULL,  
  columns = NULL,  
  ...  
)

Method toJSON():
   Usage:
   PACalculationParameters$toJSON()

Method fromJSON():
   Usage:
   PACalculationParameters$fromJSON(PACalculationParametersJson)

Method toJSONString():
   Usage:
   PACalculationParameters$toJSONString()

Method fromJSONString():
   Usage:
   PACalculationParameters$fromJSONString(PACalculationParametersJson)

Method clone(): The objects of this class are cloneable with this method.
   Usage:
   PACalculationParameters$clone(deep = FALSE)

   Arguments:
   deep  Whether to make a deep clone.

---

**Description**

PAComponent Class

**Format**

An R6Class generator object

**Public fields**

id  character [optional]
accounts list( PAIdentifier ) [optional]
benchmarks list( PAIdentifier ) [optional]
currencyisocode character [optional]
dates PADateParameters [optional]
snapshot character [optional]
name character [optional]
category character [optional]
Methods

Public methods:
- PAComponent$new()
- PAComponent$toJSON()
- PAComponent$fromJSON()
- PAComponent$toJSONString()
- PAComponent$fromJSONString()
- PAComponent$clone()

Method new():

Usage:
PAComponent$new(
  id = NULL,
  accounts = NULL,
  benchmarks = NULL,
  currencyisocode = NULL,
  dates = NULL,
  snapshot = NULL,
  name = NULL,
  category = NULL,
  ... 
)

Method toJSON():

Usage:
PAComponent$toJSON()

Method fromJSON():

Usage:
PAComponent$fromJSON(PAComponentJson)

Method toJSONString():

Usage:
PAComponent$toJSONString()

Method fromJSONString():

Usage:
PAComponent$fromJSONString(PAComponentJson)

Method clone(): The objects of this class are cloneable with this method.

Usage:
PAComponent$clone(deep = FALSE)

Arguments:
depth Whether to make a deep clone.
Description

PADateParameters Class

Format

An R6Class generator object

Public fields

startdate character [optional]
enddate character
frequency character

Methods

Public methods:

- PADateParameters$new()
- PADateParameters$toJSON()
- PADateParameters$fromJSON()
- PADateParameters$toJSONString()
- PADateParameters$fromJSONString()
- PADateParameters$clone()

Method new():

Usage:
PADateParameters$new(enddate, frequency, startdate = NULL, ...)

Method toJSON():

Usage:
PADateParameters$toJSON()

Method fromJSON():

Usage:
PADateParameters$fromJSON(PADateParametersJson)

Method toJSONString():

Usage:
PADateParameters$toJSONString()

Method fromJSONString():
Usage:
PADateParameters$fromJsonString(PADateParametersJson)

Method clone(): The objects of this class are cloneable with this method.
Usage:
PADateParameters$clone(deep = FALSE)

Arguments:
dep Whether to make a deep clone.

---

Description

PAIdentifier Class

Format

An R6Class generator object

Public fields

id character
holdingsmode character [optional]

Methods

Public methods:
- `PAIdentifier$new()`
- `PAIdentifier$toJSON()`
- `PAIdentifier$fromJson()`
- `PAIdentifier$toJSONString()`
- `PAIdentifier$fromJsonString()`
- `PAIdentifier$clone()`

Method new():

Usage:
PAIdentifier$new(id, holdingsmode = NULL, ...)

Method toJSON():

Usage:
PAIdentifier$toJSON()

Method fromJSON():
**Usage:**

```r
PAIdentifier$fromJSON( PAIdentifierJson )
```

**Method** `toJSONString()`:

**Usage:**

```r
PAIdentifier$toJSONString()
```

**Method** `fromJSONString()`:

**Usage:**

```r
PAIdentifier$fromJSONString( PAIdentifierJson )
```

**Method** `clone()`: The objects of this class are cloneable with this method.

**Usage:**

```r
PAIdentifier$clone( deep = FALSE )
```

**Arguments:**

- `deep` Whether to make a deep clone.

---

**Description**

**SPARCCalculationParameters Class**

**Format**

An R6Class generator object

**Public fields**

- `componentid` character
- `accounts` list (SPARIdentifier) [optional]
- `benchmark` SPARIdentifier [optional]
- `dates` SPARDateParameters [optional]

**Methods**

**Public methods:**

- `SPARCcalculationParameters$new()`
- `SPARCcalculationParameters$toJSON()`
- `SPARCcalculationParameters$fromJSON()`
- `SPARCcalculationParameters$toJSONString()`
- `SPARCcalculationParameters$fromJSONString()`
• SPARCalculationParameters$clone()

Method new():
Usage:
SPARCalculationParameters$new(
  componentid,
  accounts = NULL,
  benchmark = NULL,
  dates = NULL,
  ...
)

Method toJSON():
Usage:
SPARCalculationParameters$toJSON()

Method fromJSON():
Usage:
SPARCalculationParameters$fromJSON(SPARCalculationParametersJson)

Method toJSONString():
Usage:
SPARCalculationParameters$toJSONString()

Method fromJSONString():
Usage:
SPARCalculationParameters$fromJSONString(SPARCalculationParametersJson)

Method clone(): The objects of this class are cloneable with this method.
Usage:
SPARCalculationParameters$clone(deep = FALSE)
Arguments:
deep Whether to make a deep clone.

SPARDateParameters

Description
SPARDateParameters Class

Format
An R6Class generator object
SPARDateParameters

Public fields

startdate character
enddate character
frequency character

Methods

Public methods:

• SPARDateParameters$new()
• SPARDateParameters$toJSON()
• SPARDateParameters$fromJSON()
• SPARDateParameters$toJSONString()
• SPARDateParameters$fromJSONString()
• SPARDateParameters$clone()

Method new():

Usage:
SPARDateParameters$new(startdate, enddate, frequency, ...)

Method toJSON():

Usage:
SPARDateParameters$toJSON()

Method fromJSON():

Usage:
SPARDateParameters$fromJSON(SPARDateParametersJson)

Method toJSONString():

Usage:
SPARDateParameters$toJSONString()

Method fromJSONString():

Usage:
SPARDateParameters$fromJSONString(SPARDateParametersJson)

Method clone(): The objects of this class are cloneable with this method.

Usage:
SPARDateParameters$clone(deep = FALSE)

Arguments:
dee...
Description

SPARIdentifier Class

Format

An R6Class generator object

Public fields

id character

returntype character [optional]

prefix character [optional]

Methods

Public methods:

• SPARIdentifier$new()
• SPARIdentifier$toJSON()
• SPARIdentifier$fromJSON()
• SPARIdentifier$toJSONString()
• SPARIdentifier$fromJSONString()
• SPARIdentifier$clone()

Method new():

Usage:
SPARIdentifier$new(id, returntype = NULL, prefix = NULL, ...)

Method toJSON():

Usage:
SPARIdentifier$toJSON()

Method fromJSON():

Usage:
SPARIdentifier$fromJSON(SPARIdentifierJson)

Method toJSONString():

Usage:
SPARIdentifier$toJSONString()

Method fromJSONString():
Usage:
SPARIdentifier$fromJSONString(SPARIdentifierJson)

Method clone(): The objects of this class are cloneable with this method.
Usage:
SPARIdentifier$clone(deep = FALSE)
Arguments:
deep Whether to make a deep clone.

---

UtilityApi Utility operations

Description
factset.analyticsapi.engines

Format
An R6Class generator object

Methods

**GetByUrl**  *Url of the GET endpoint*  This function can be used to fetch data from any Get endpoint.

@param url character

- @returnType raw
- On encountering errors, an error of subclass ApiException will be thrown.
- status code : 200 | Expected response once the request is successful. Response body will contain the data.
- return type : raw
- response headers :

  - X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-DataDirect-Request-Key FactSet’s request key header.

  - status code : 400 | Invalid identifier provided.
  - response headers :

  - X-FactSet-Api-Request-Key Key to uniquely identify an Analytics API request. Only available after successful authentication.
  - X-DataDirect-Request-Key FactSet’s request key header.

  - status code : 401 | Missing or invalid authentication.
  - response headers :
UtilityApi

X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-DataDirect-Request-Key  FactSet’s request key header.

- status code : 403 | User is forbidden with current credentials
- response headers :

X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-DataDirect-Request-Key  FactSet’s request key header.

- status code : 406 | Unsupported Accept header.
- response headers :

X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-DataDirect-Request-Key  FactSet’s request key header.

- status code : 500 | Server error. Log the X-DataDirect-Request-Key header to assist in troubleshooting.
- response headers :

X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-DataDirect-Request-Key  FactSet’s request key header.

- status code : 503 | Request timed out. Retry the request in sometime.
- response headers :

X-FactSet-Api-Request-Key  Key to uniquely identify an Analytics API request. Only available after successful authentication.
X-DataDirect-Request-Key  FactSet’s request key header.

Public fields

apiClient  Handles the client-server communication.

Methods

Public methods:

- UtilityApi$new()
- UtilityApi$GetByUrl()
- UtilityApi$GetByUrlWithHttpInfo()
- UtilityApi$clone()

Method new():

Usage:
UtilityApi$new(apiClient)

**Method** GetByUrl():

*Usage:*
UtilityApi$GetByUrl(url, ...)

**Method** GetByUrlWithHttpInfo():

*Usage:*
UtilityApi$GetByUrlWithHttpInfo(url, ...)

**Method** clone(): The objects of this class are cloneable with this method.

*Usage:*
UtilityApi$clone(deep = FALSE)

*Arguments:*
depth Whether to make a deep clone.

**Examples**

```r
## Not run:
library(factset.analyticsapi.engines)
var.url <- url # character | Url of the GET endpoint

# Get by url
api.instance <- UtilityApi$new()

#Configure HTTP basic authorization: Basic
# provide your username in the user-serial format
api.instance$apiClient$username <- '<user-serial>

# provide your api key generated using the developer portal
api.instance$apiClient$password <- '<api_key>

result <- tryCatch(
    api.instance$GetByUrl(url = var.url),
    ApiException = function(ex) ex
)

# In case of error, print the error object
if(!is.null(result$ApiException)) {
    cat(result$ApiException$toString())
} else {
    # deserialized response object
    response.object <- result$content
    # response headers
    response.headers <- result$response$headers
    # response status code
    response.status.code <- result$response$status_code
}

## End(Not run)
```
**VaultCalculationParameters**

*VaultCalculationParameters*

---

**Description**

VaultCalculationParameters Class

**Format**

An R6Class generator object

**Public fields**

- componentid character
- account *VaultIdentifier*
- dates *VaultDateParameters* [optional]
- configid character

**Methods**

- **Public methods:**
  - *VaultCalculationParameters$new()
  - *VaultCalculationParameters$toJSON()
  - *VaultCalculationParameters$fromJSON()
  - *VaultCalculationParameters$toJSONString()
  - *VaultCalculationParameters$fromJSONString()
  - *VaultCalculationParameters$clone()

- **Method** `new()`:
  
  *Usage:*
  
  ```
  VaultCalculationParameters$new(
    componentid, account, configid,
    dates = NULL, ...
  )
  ```

- **Method** `toJSON()`:
  
  *Usage:*
  
  ```
  VaultCalculationParameters$toJSON()
  ```

- **Method** `fromJSON()`:

  *Usage:*

  ```
  VaultCalculationParameters$fromJSON()
  ```
VaultComponent

VaultComponent

VaultComponent

VaultComponent

Method toJSONString():

Usage:
VaultCalculationParameters$toJSONString()

Method fromJSONString():

Usage:
VaultCalculationParameters$fromJSONString(VaultCalculationParametersJson)

Method clone(): The objects of this class are cloneable with this method.

Usage:
VaultCalculationParameters$clone(deep = FALSE)

Arguments:

depth Whether to make a deep clone.

VaultComponent

Description

VaultComponent Class

Format

An R6Class generator object

Public fields

id character [optional]
account VaultIdentifier [optional]
benchmark VaultIdentifier [optional]
currencyisocode character [optional]
dates VaultDateParameters [optional]
snapshot character [optional]
name character [optional]
category character [optional]
Methods

Public methods:

• `VaultComponent$new()`  
• `VaultComponent$toJSON()`  
• `VaultComponent$fromJSON()`  
• `VaultComponent$toJSONString()`  
• `VaultComponent$fromJSONString()`  
• `VaultComponent$clone()`

Method `new()`:

Usage:

```
VaultComponent$new(
    id = NULL,
    account = NULL,
    benchmark = NULL,
    currencyisocode = NULL,
    dates = NULL,
    snapshot = NULL,
    name = NULL,
    category = NULL,
    ...
)
```

Method `toJSON()`:

Usage:

```
VaultComponent$toJSON()
```

Method `fromJSON()`:

Usage:

```
VaultComponent$fromJSON(VaultComponentJson)
```

Method `toJSONString()`:

Usage:

```
VaultComponent$toJSONString()
```

Method `fromJSONString()`:

Usage:

```
VaultComponent$fromJSONString(VaultComponentJson)
```

Method `clone()`: The objects of this class are cloneable with this method.

Usage:

```
VaultComponent$clone(deep = FALSE)
```

Arguments:

depth Whether to make a deep clone.
VaultConfiguration

Description
VaultConfiguration Class

Format
An R6Class generator object

Public fields

name character [optional]
accounts named list( ConfigurationAccount ) [optional]

Methods

Public methods:

• VaultConfiguration$new()
• VaultConfiguration$toJSON()
• VaultConfiguration$fromJSON()
• VaultConfiguration$toJSONString()
• VaultConfiguration$fromJSONString()
• VaultConfiguration$clone()

Method new():
Usage:
VaultConfiguration$new(name = NULL, accounts = NULL, ...)

Method toJSON():
Usage:
VaultConfiguration$toJSON()

Method fromJSON():
Usage:
VaultConfiguration$fromJSON(VaultConfigurationJson)

Method toJSONString():
Usage:
VaultConfiguration$toJSONString()

Method fromJSONString():
Usage:
 VaultConfigurationSummary

**Method** `clone()`: The objects of this class are cloneable with this method.

*Usage:*

`VaultConfigurationSummary$clone(deep = FALSE)`

*Arguments:*

depth Whether to make a deep clone.

---

**Description**

*VaultConfigurationSummary Class*

**Format**

An R6Class generator object

**Public fields**

name character [optional]

**Methods**

**Public methods:**

- `VaultConfigurationSummary$new()`
- `VaultConfigurationSummary$toJSON()`
- `VaultConfigurationSummary$fromJSON()`
- `VaultConfigurationSummary$toJSONString()`
- `VaultConfigurationSummary$fromJSONString()`
- `VaultConfigurationSummary$clone()`

**Method** `new()`:

*Usage:*

`VaultConfigurationSummary$new(name = NULL, ...)`

**Method** `toJSON()`:

*Usage:*

`VaultConfigurationSummary$toJSON()`

**Method** `fromJSON()`:

*Usage:*

`VaultConfigurationSummary$fromJSON(VaultConfigurationSummaryJson)`
Method toJSONString():

Usage:
VaultConfigurationSummary$toJSONString()

Method fromJSONString():

Usage:
VaultConfigurationSummary$fromJSONString(VaultConfigurationSummaryJson)

Method clone(): The objects of this class are cloneable with this method.

Usage:
VaultConfigurationSummary$clone(deep = FALSE)

Arguments:
depth Whether to make a deep clone.

Description

VaultDateParameters Class

Format

An R6Class generator object

Public fields

startdate character [optional]
enddate character
frequency character

Methods

Public methods:

• VaultDateParameters$new()
• VaultDateParameters$toJSON()
• VaultDateParameters$fromJSON()
• VaultDateParameters$toJSONString()
• VaultDateParameters$fromJSONString()
• VaultDateParameters$clone()

Method new():

Usage:
VaultDateParameters$new(enddate, frequency, startdate = NULL, ...)

Method toJSON():

Usage:
VaultDateParameters$toJSON()

Method fromJSON():

Usage:
VaultDateParameters$fromJson(VaultDateParametersJson)

Method toJSONString():

Usage:
VaultDateParameters$toJSONString()

Method fromJSONString():

Usage:
VaultDateParameters$fromJsonString(VaultDateParametersJson)

Method clone(): The objects of this class are cloneable with this method.

Usage:
VaultDateParameters$clone(deep = FALSE)

Arguments:

depth  Whether to make a deep clone.

VaultIdentifier  VaultIdentifier

Description

VaultIdentifier Class

Format

An R6Class generator object

Public fields

id  character
Methods

Public methods:

• `VaultIdentifier$new()`
• `VaultIdentifier$toJSON()`
• `VaultIdentifier$fromJSON()`
• `VaultIdentifier$toJSONString()`
• `VaultIdentifier$fromJSONString()`
• `VaultIdentifier$clone()`

**Method new():**

*Usage:*

`VaultIdentifier$new(id, ...)`

**Method toJSON():**

*Usage:*

`VaultIdentifier$toJSON()`

**Method fromJSON():**

*Usage:*

`VaultIdentifier$fromJSON(VaultIdentifierJson)`

**Method toJSONString():**

*Usage:*

`VaultIdentifier$toJSONString()`

**Method fromJSONString():**

*Usage:*

`VaultIdentifier$fromJSONString(VaultIdentifierJson)`

**Method clone():** The objects of this class are cloneable with this method.

*Usage:*

`VaultIdentifier$clone(deep = FALSE)`

*Arguments:*

depth Whether to make a deep clone.
Index

AccountDirectories, 3, 4
AccountsApi, 4
ApiClient, 7
ApiException, 9
ApiResponse, 10
Calculation, 11, 17
CalculationsApi, 12
CalculationStatus, 14, 23
CalculationStatusSummary, 16, 25
CalculationUnitStatus, 24, 26
Column, 28, 29
ColumnsApi, 29
ColumnStatistic, 35, 36
ColumnStatisticsApi, 36
ColumnSummary, 31, 39
ComponentAccount, 40
ComponentBenchmark, 42
ComponentsApi, 43
ComponentSummary, 45, 47, 50, 57
ConfigurationAccount, 58, 119
ConfigurationsApi, 60
CurrenciesApi, 66
Currency, 66, 69
DateParametersSummary, 70, 72, 74
DatesApi, 72
DocumentDirectories, 78, 80, 82, 84
DocumentsApi, 80
FrequenciesApi, 88
Frequency, 89–91, 96
Group, 97, 98
GroupsApi, 98
PACalculationColumn, 101, 104
PACalculationGroup, 102, 104
PACalculationParameters, 11, 104
PAComponent, 43, 105
PADateParameters, 104, 105, 107
PAIdentifier, 104, 105, 108
SPARCalculationParameters, 11, 109
SPARDateParameters, 109, 110
SPARIdentifier, 109, 112
UtilityApi, 113
VaultCalculationParameters, 11, 116
VaultComponent, 49, 117
VaultConfiguration, 60, 119
VaultConfigurationSummary, 62, 120
VaultDateParameters, 116, 117, 121
VaultIdentifier, 116, 117, 122