
cognite-sdk-python Documentation

Release 0.12.15

Erlend Vollset

Feb 12, 2019

Contents

1	Overview	3
1.1	Cognite Client	3
1.2	Responses	4
1.3	Exceptions	5
2	API	7
2.1	Assets	7
2.2	Datapoints	11
2.3	Events	17
2.4	Files	21
2.5	Login	25
2.6	Raw	26
2.7	Tagmatching	28
2.8	Time Series	29
3	Experimental	33
3.1	Analytics	33
3.2	Time Series	37
3.3	Datapoints	39
3.4	Sequences	40
4	Data Transfer Service	45
	Python Module Index	49

Python SDK to ensure excellent user experience for developers and data scientists working with the Cognite Data Platform.

To install this package run the following command

```
pip install cognite-sdk
```


1.1 Cognite Client

```
class cognite.CogniteClient (api_key: str = None, project: str = None, base_url: str = None,  
                             num_of_retries: int = None, num_of_workers: int = None, headers:  
                             Dict[str, str] = None, cookies: Dict[str, str] = None, timeout: int =  
                             None, debug: bool = None)
```

Main entrypoint into Cognite Python SDK.

All services are made available through this object. See examples below.

Parameters

- **api_key** (*str*) – API key
- **project** (*str*) – Project. Defaults to project of given API key.
- **base_url** (*str*) – Base url to send requests to. Defaults to “<https://api.cognitedata.com>”
- **num_of_retries** (*int*) – Number of times to retry failed requests. Defaults to 5. Will only retry status codes 401, 429, 500, 502, and 503.
- **num_of_workers** (*int*) – Number of workers to spawn when parallelizing data fetching. Defaults to 10.
- **cookies** (*Dict*) – Cookies to append to all requests. Defaults to {}
- **headers** (*Dict*) – Additional headers to add to all requests. Defaults are: {“api-key”: self.api_key, “content-type”: “application/json”, “accept”: “application/json”}
- **timeout** (*int*) – Timeout on requests sent to the api. Defaults to 60 seconds.
- **debug** (*bool*) – Configures logger to log extra request details to stdout.

Examples

The CogniteClient is instantiated and used like this. This example assumes that the environment variable COGNITE_API_KEY has been set:

```
from cognite import CogniteClient
client = CogniteClient()
res = client.time_series.get_time_series()
print(res.to_pandas())
```

Certain experimental features are made available through this client as follows:

```
from cognite import CogniteClient
client = CogniteClient()
res = client.experimental.analytics.models.get_models()
print(res)
```

Default configurations may be set using the following environment variables:

```
export COGNITE_API_KEY = <your-api-key>
export COGNITE_BASE_URL = http://<host>:<port>
export COGNITE_NUM_RETRIES = <number-of-retries>
export COGNITE_NUM_WORKERS = <number-of-workers>
export COGNITE_TIMEOUT = <num-of-seconds>
```

get (*url: str, params: Dict[str, Any] = None, headers: Dict[str, Any] = None*)
Perform a GET request to a path in the API.

Comes in handy if the endpoint you want to reach is not currently supported by the SDK.

post (*url: str, body: Dict[str, Any], params: Dict[str, Any] = None, use_gzip: bool = True, headers: Dict[str, Any] = None*)
Perform a POST request to a path in the API.

Comes in handy if the endpoint you want to reach is not currently supported by the SDK.

put (*url: str, body: Dict[str, Any] = None, headers: Dict[str, Any] = None*)
Perform a PUT request to a path in the API.

Comes in handy if the endpoint you want to reach is not currently supported by the SDK.

delete (*url: str, params: Dict[str, Any] = None, headers: Dict[str, Any] = None*)
Perform a DELETE request to a path in the API.

Comes in handy if the endpoint you want to reach is not currently supported by the SDK.

1.2 Responses

class `cognite.client._api_client.CogniteResponse` (*internal_representation*)
Bases: `object`

Cognite Response class

All responses inherit from this class.

Examples

All responses are pretty-printable:


```

from cognite import CogniteClient

client = CogniteClient()
res = client.assets.get_assets(limit=1)

print(res)

```

All endpoints which support paging have an `autopaging` flag which may be set to true in order to sequentially fetch all resources. If for some reason, you want to do this manually, you may use the `next_cursor()` method on the response object. Here is an example of that:

```

from cognite import CogniteClient

client = CogniteClient()

asset_list = []

cursor = None
while True:
    res = client.assets.get_assets(cursor=cursor)
    asset_list.extend(res.to_json())
    cursor = res.next_cursor()
    if cursor is None:
        break

print(asset_list)

```

next_cursor()

Returns next cursor to use for paging through results. Returns None if there are no more results.

previous_cursor()

Returns previous cursor to use for paging through results. Returns None if there are no more results.

to_json()

Returns data as a json object

class `cognite.client._api_client.CogniteCollectionResponse` (*internal_representation*)

Bases: `cognite.client._api_client.CogniteResponse`

Cognite Collection Response class

All collection responses inherit from this class. Collection responses are subscriptable and iterable.

to_json()

Returns data as a json object

1.3 Exceptions

exception `cognite.client.exceptions.APIError` (*message, code=None, x_request_id=None, extra=None*)

Bases: Exception

Cognite API Error

Raised if a given request fails.

Parameters

- **message** (*str*) – The error message produced by the API

- **code** (*int*) – The error code produced by the failure
- **x_request_id** (*str*) – The request-id generated for the failed request.
- **extra** (*Dict*) – A dict of any additional information.

Examples

Catching an API-error and handling it based on the error code:

```
from cognite import CogniteClient, APIError

client = CogniteClient()

try:
    client.login.status()
except APIError as e:
    if e.code == 401:
        print("You are not authorized")
    elif e.code == 400:
        print("Something is wrong with your request")
    elif e.code == 500:
        print("Something went terribly wrong. Here is the request-id: {}".
↪format(e.x_request_id))
        print("The message returned from the API: {}".format(e.message))
```

2.1 Assets

2.1.1 Client

class `cognite.client.stable.assets.AssetsClient` (**kwargs)

delete_assets (*asset_ids: List[int]*) → None

Delete a list of assets.

Parameters **asset_ids** (*list [int]*) – List of IDs of assets to delete.

Returns None

Examples

You can delete an asset like this:

```
client = CogniteClient()
res = client.assets.delete_assets([123])
```

get_asset (*asset_id*) → `cognite.client.stable.assets.AssetResponse`

Returns the asset with the provided assetId.

Parameters **asset_id** (*int*) – The asset id of the top asset to get.

Returns A data object containing the requested assets with several getter methods with different output formats.

Return type *stable.assets.AssetResponse*

Examples

You can fetch a single asset like this:

```
client = CogniteClient()
res = client.assets.get_asset(asset_id=123)
print(res)
```

get_asset_subtree (*asset_id*, *depth=None*, ***kwargs*) → *cognite.client.stable.assets.AssetListResponse*
Returns asset subtree of asset with provided assetId.

Parameters

- **asset_id** (*int*) – The asset id of the top asset to get.
- **depth** (*int*) – Get subassets this many levels below the top asset.

Keyword Arguments

- **limit** (*int*) – The maximum number of assets to be returned.
- **cursor** (*str*) – Cursor to use for paging through results.

Returns A data object containing the requested assets with several getter methods with different output formats.

Return type *stable.assets.AssetListResponse*

Examples

You can fetch an asset subtree like this:

```
client = CogniteClient()
res = client.assets.get_asset_subtree(asset_id=123, depth=)
print(res.to_pandas())
```

get_assets (*name=None*, *path=None*, *description=None*, *metadata=None*, *depth=None*, *fuzziness=None*, ***kwargs*) → *cognite.client.stable.assets.AssetListResponse*
Returns assets matching provided description.

Parameters

- **name** (*str*) – The name of the asset(s) to get.
- **path** (*str*) – The path of the subtree to search in.
- **description** (*str*) – Search query.
- **metadata** (*dict*) – The metadata values used to filter the results.
- **depth** (*int*) – Get sub assets up to this many levels below the specified path.
- **fuzziness** (*int*) – The degree of fuzziness in the name matching.

Keyword Arguments

- **autopaging** (*bool*) – Whether or not to automatically page through results. If set to true, limit will be disregarded. Defaults to False.
- **limit** (*int*) – The maximum number of assets to be returned.
- **cursor** (*str*) – Cursor to use for paging through results.

Returns A data object containing the requested assets with several getter methods with different output formats.

Return type `stable.assets.AssetListResponse`

Examples

You can fetch all assets with a maximum depth of three like this:

```
client = CogniteClient()
res = client.assets.get_assets(depth=3, autopaging=True)
print(res.to_pandas())
```

post_assets (*assets*: `List[cognite.client.stable.assets.Asset]` → `cognite.client.stable.assets.AssetListResponse`)

Insert a list of assets.

Parameters **assets** (`list[stable.assets.Asset]`) – List of asset data transfer objects.

Returns A data object containing the posted assets with several getter methods with different output formats.

Return type `stable.assets.AssetListResponse`

Examples

Posting an asset:

```
from cognite.client.stable.assets import Asset

client = CogniteClient()

my_asset = Asset("myasset")
assets_to_post = [my_asset]
res = client.assets.post_assets(assets_to_post)
print(res)
```

search_for_assets (*name=None, description=None, query=None, metadata=None, asset_subtrees=None, min_created_time=None, max_created_time=None, min_last_updated_time=None, max_last_updated_time=None, **kwargs*) → `cognite.client.stable.assets.AssetListResponse`

Search for assets.

Parameters

- **name** – Prefix and fuzzy search on name.
- **str** (*description*) – Prefix and fuzzy search on description.
- **query** (*str*) – Search on name and description using wildcard search on each of the words (separated by spaces). Retrieves results where at least one word must match. Example: ‘some other’
- **metadata** (*dict*) – Filter out assets that do not match these metadata fields and values (case-sensitive). Format is {“key1”:”value1”,”key2”:”value2”}.
- **asset_subtrees** (`List[int]`) – Filter out assets that are not linked to assets in the subtree rooted at these assets. Format is [12,345,6,7890].

- **min_created_time** (*str*) – Filter out assets with createdTime before this. Format is milliseconds since epoch.
- **max_created_time** (*str*) – Filter out assets with createdTime after this. Format is milliseconds since epoch.
- **min_last_updated_time** (*str*) – Filter out assets with lastUpdatedtime before this. Format is milliseconds since epoch.
- **max_last_updated_time** (*str*) – Filter out assets with lastUpdatedtime after this. Format is milliseconds since epoch.

Keyword Arguments

- **sort** (*str*) – Field to be sorted.
- **dir** (*str*) – Sort direction (desc or asc)
- **limit** (*int*) – Return up to this many results. Max is 1000, default is 25.
- **offset** (*int*) – Offset from the first result. Sum of limit and offset must not exceed 1000. Default is 0.
- **boost_name** (*str*) – Whether or not boosting name field. This option is test_experimental and can be changed.

Returns stable.assets.AssetListResponse.

Examples

Searching for assets:

```
client = CogniteClient()
res = client.assets.search_for_assets(name="myasset")
print(res)
```

2.1.2 Data Objects

class cognite.client.stable.assets.**Asset** (*name, parent_id=None, description=None, meta-data=None, ref_id=None, parent_name=None, parent_ref_id=None*)

Bases: object

Data transfer object for assets.

Parameters

- **name** (*str*) – Name of asset. Often referred to as tag.
- **parent_id** (*int*) – ID of parent asset, if any.
- **description** (*str*) – Description of asset.
- **metadata** (*dict*) – Custom , application specific metadata. String key -> String Value.
- **ref_id** (*str*) – Reference ID used only in post request to disambiguate references to duplicate names.
- **parent_name** (*str*) – Name of parent, this parent must exist in the same POST request.
- **parent_ref_id** (*str*) – Reference ID of parent, to disambiguate if multiple nodes have the same name.

```

class cognite.client.stable.assets.AssetListResponse (internal_representation)
    Bases: cognite.client._api_client.CogniteCollectionResponse

    Assets Response Object

    next_cursor ()
        Returns next cursor to use for paging through results. Returns None if there are no more results.

    previous_cursor ()
        Returns previous cursor to use for paging through results. Returns None if there are no more results.

    to_json ()
        Returns data as a json object

    to_pandas ()
        Returns data as a pandas dataframe

class cognite.client.stable.assets.AssetResponse (internal_representation)
    Bases: cognite.client._api_client.CogniteResponse

    next_cursor ()
        Returns next cursor to use for paging through results. Returns None if there are no more results.

    previous_cursor ()
        Returns previous cursor to use for paging through results. Returns None if there are no more results.

    to_json ()
        Returns data as a json object

    to_pandas ()
        Returns data as a pandas dataframe

```

2.2 Datapoints

2.2.1 Client

```

class cognite.client.stable.datapoints.DatapointsClient (**kwargs)

    get_datapoints (name, start, end=None, aggregates=None, granularity=None, **kwargs) → cog-
        nite.client.stable.datapoints.DatapointsResponse
        Returns a DatapointsObject containing a list of datapoints for the given query.

        This method will automate paging for the user and return all data for the given time period.

```

Parameters

- **name** (*str*) – The name of the timeseries to retrieve data for.
- **start** (*Union[str, int, datetime]*) – Get datapoints after this time. Format is N[timeunit]-ago where timeunit is w,d,h,m,s. E.g. ‘2d-ago’ will get everything that is up to 2 days old. Can also send time in ms since epoch or a datetime object which will be converted to ms since epoch UTC.
- **end** (*Union[str, int, datetime]*) – Get datapoints up to this time. Same format as for start.
- **aggregates** (*list*) – The list of aggregate functions you wish to apply to the data. Valid aggregate functions are: ‘average/avg, max, min, count, sum, interpolation/int, stepinterpolation/step’.

- **granularity** (*str*) – The granularity of the aggregate values. Valid entries are : ‘day/d, hour/h, minute/m, second/s’, or a multiple of these indicated by a number as a prefix e.g. ‘12hour’.

Keyword Arguments

- **workers** (*int*) – Number of download workers to run in parallel. Defaults to 10.
- **include_outside_points** (*bool*) – No description
- **protobuf** (*bool*) – Download the data using the binary protobuf format. Only applicable when getting raw data. Defaults to True.
- **limit** (*str*) – Max number of datapoints to return. If limit is specified, this method will not automate paging and will return a maximum of 100,000 dps.

Returns A data object containing the requested data with several getter methods with different output formats.

Return type *stable.datapoints.DatapointsResponse*

Examples

Getting the last 3 days of raw datapoints for a given time series:

```
client = CogniteClient()
res = client.datapoints.get_datapoints(name="my_ts", start="3d-ago")
print(res.to_pandas())
```

get_datapoints_frame (*time_series, aggregates, granularity, start, end=None, **kwargs*) → pandas.core.frame.DataFrame

Returns a pandas dataframe of datapoints for the given timeseries all on the same timestamps.

This method will automate paging for the user and return all data for the given time period.

Parameters

- **time_series** (*list*) – The list of timeseries names to retrieve data for. Each timeseries can be either a string containing the timeseries or a dictionary containing the names of the timeseries and a list of specific aggregate functions.
- **aggregates** (*list*) – The list of aggregate functions you wish to apply to the data for which you have not specified an aggregate function. Valid aggregate functions are: ‘average/avg, max, min, count, sum, interpolation/int, stepinterpolation/step’.
- **granularity** (*str*) – The granularity of the aggregate values. Valid entries are : ‘day/d, hour/h, minute/m, second/s’, or a multiple of these indicated by a number as a prefix e.g. ‘12hour’.
- **start** (*Union[str, int, datetime]*) – Get datapoints after this time. Format is N[timeunit]-ago where timeunit is w,d,h,m,s. E.g. ‘2d-ago’ will get everything that is up to 2 days old. Can also send time in ms since epoch or a datetime object which will be converted to ms since epoch UTC.
- **end** (*Union[str, int, datetime]*) – Get datapoints up to this time. Same format as for start.

Keyword Arguments

- **limit** (*str*) – Max number of rows to return. If limit is specified, this method will not automate paging and will return a maximum of 100,000 rows.

- **workers** (*int*) – Number of download workers to run in parallel. Defaults to 10.

Returns A pandas dataframe containing the datapoints for the given timeseries. The datapoints for all the timeseries will all be on the same timestamps.

Return type pandas.DataFrame

Examples

Get a dataframe of aggregated time series data:

```
client = CogniteClient()

res = client.datapoints.get_datapoints_frame(time_series=["ts1", "ts2"],
                                             aggregates=["avg"], granularity="30s", start="1w-ago")

print(res)
```

The `timeseries` parameter can take a list of strings and/or dicts on the following formats. This is useful for specifying aggregate functions on a per time series level:

```
Using strings:
    ['<timeseries1>', '<timeseries2>']

Using dicts:
    [{'name': '<timeseries1>', 'aggregates': ['<aggfunc1>', '<aggfunc2>']},
     {'name': '<timeseries2>', 'aggregates': []}]

Using both:
    ['<timeseries1>', {'name': '<timeseries2>', 'aggregates': ['<aggfunc1>', '↪<aggfunc2>']}]
```

get_latest (*name*, *before=None*) → `cognite.client.stable.datapoints.LatestDatapointResponse`
Returns a `LatestDatapointObject` containing the latest datapoint for the given timeseries.

Parameters **name** (*str*) – The name of the timeseries to retrieve data for.

Returns A data object containing the requested data with several getter methods with different output formats.

Return type `stable.datapoints.LatestDatapointsResponse`

Examples

Get the latest datapoint from a time series before time x:

```
client = CogniteClient()
x = 1514761200000
client.datapoints.get_latest(name="my_ts", before=x)
```

get_multi_time_series_datapoints (*datapoints_queries: List[cognite.client.stable.datapoints.DatapointsQuery]*,
start, *end=None*, *aggregates=None*,
granularity=None, ***kwargs*) → `cog-`
`nite.client.stable.datapoints.DatapointsResponseIterator`

Returns a list of `DatapointsObjects` each of which contains a list of datapoints for the given timeseries.

This method will automate paging for the user and return all data for the given time period(s).

Parameters

- **datapoints_queries** (*list* [*stable.datapoints.DatapointsQuery*]) – The list of DatapointsQuery objects specifying which timeseries to retrieve data for.
- **start** (*Union*[*str*, *int*, *datetime*]) – Get datapoints after this time. Format is N[timeunit]-ago where timeunit is w,d,h,m,s. E.g. ‘2d-ago’ will get everything that is up to 2 days old. Can also send time in ms since epoch or a datetime object which will be converted to ms since epoch UTC.
- **end** (*Union*[*str*, *int*, *datetime*]) – Get datapoints up to this time. Same format as for start.
- **aggregates** (*list*, *optional*) – The list of aggregate functions you wish to apply to the data. Valid aggregate functions are: ‘average/avg, max, min, count, sum, interpolation/int, stepinterpolation/step’.
- **granularity** (*str*) – The granularity of the aggregate values. Valid entries are : ‘day/d, hour/h, minute/m, second/s’, or a multiple of these indicated by a number as a prefix e.g. ‘12hour’.

Keyword Arguments **include_outside_points** (*bool*) – No description.

Returns An iterator which iterates over *stable.datapoints.DatapointsResponse* objects.

Return type *stable.datapoints.DatapointsResponseIterator*

live_data_generator (*name*, *update_frequency=1*)

Generator function which continuously polls latest datapoint of a timeseries and yields new datapoints.

Parameters

- **name** (*str*) – Name of timeseries to get latest datapoints for.
- **update_frequency** (*float*) – Frequency to pull for data in seconds.

Yields *dict* – Dictionary containing timestamp and value of latest datapoint.

post_datapoints (*name*, *datapoints: List*[*cognite.client.stable.datapoints.Datapoint*]) → None

Insert a list of datapoints.

Parameters

- **name** (*str*) – Name of timeseries to insert to.
- **datapoints** (*List*[*stable.datapoints.Datapoint*]) – List of datapoint data transfer objects to insert.

Returns None

Examples

Posting some dummy datapoints:

```
from cognite.client.stable.datapoints import Datapoint

client = CogniteClient()

start = 1514761200000
my_dummy_data = [Datapoint(timestamp=start+off, value=off) for off in
↳ range(100)]
client.datapoints.post_datapoints(ts_name, my_dummy_data)
```

post_datapoints_frame (*dataframe*) → None

Write a dataframe. The dataframe must have a 'timestamp' column with timestamps in milliseconds since epoch. The names of the remaining columns specify the names of the time series to which column contents will be written. Said time series must already exist.

Parameters **dataframe** (*pandas.DataFrame*) – Pandas DataFrame Object containing the time series.

Returns None

Examples

Post a dataframe with white noise:

```
client = CogniteClient()
ts_name = 'NOISE'

start = datetime(2018, 1, 1)
# The scaling by 1000 is important: timestamp() returns seconds
x = [(start + timedelta(days=d)).timestamp() * 1000 for d in range(100)]
y = np.random.normal(0, 1, 100)

# The time column must be called precisely 'timestamp'
df = pd.DataFrame({'timestamp': x, ts_name: y})

client.datapoints.post_datapoints_frame(df)
```

post_multi_time_series_datapoints (*timeseries_with_datapoints:*

List[cognite.client.stable.datapoints.TimeseriesWithDatapoints],
**kwargs) → None

Insert data into multiple timeseries.

Parameters **timeseries_with_datapoints** (*List[stable.datapoints.TimeseriesWithDatapoints]*) – The timeseries with data to insert.

Keyword Arguments **use_gzip** (*bool*) – Whether or not to gzip the request. Defaults to True.

Returns None

Examples

Posting some dummy datapoints to multiple time series. This example assumes that the time series have already been created:

```
from cognite.client.stable.datapoints import TimeseriesWithDatapoints, \
↳Datapoint

start = 1514761200000
my_dummy_data_1 = [Datapoint(timestamp=ms, value=i) for i, ms in range(start, \
↳start+100)]
ts_with_datapoints_1 = TimeseriesWithDatapoints(name="ts1", datapoints=my_
↳dummy_data_1)

start = 1503331800000
my_dummy_data_2 = [Datapoint(timestamp=ms, value=i) for i, ms in range(start, \
↳start+100)]
```

(continues on next page)

(continued from previous page)

```

ts_with_datapoints_2 = TimeSeriesWithDatapoints(name="ts2", datapoints=my_
↳dummy_data_2)

my_dummy_data = [ts_with_datapoints_1, ts_with_datapoints_2]

client = CogniteClient()
res = client.datapoints.post_multi_time_series_datapoints(my_dummy_data)

```

2.2.2 Data Objects

class `cognite.client.stable.datapoints.Datapoint` (*timestamp, value*)

Bases: `object`

Data transfer object for datapoints.

Parameters

- **timestamp** (*int, datetime*) – The data timestamp in milliseconds since the epoch (Jan 1, 1970) or as a datetime object.
- **value** (*string*) – The data value, Can be string or numeric depending on the metric.

class `cognite.client.stable.datapoints.DatapointsQuery` (*name, aggregates=None, granularity=None, start=None, end=None, limit=None*)

Bases: `object`

Data Query Object for Datapoints.

Parameters

- **name** (*str*) – Unique name of the time series.
- **aggregates** (*list*) – The aggregate functions to be returned. Use default if null. An empty string must be sent to get raw data if the default is a set of aggregate functions.
- **granularity** (*str*) – The granularity size and granularity of the aggregates.
- **start** (*str, int, datetime*) – Get datapoints after this time. Format is N[timeunit]-ago where timeunit is w,d,h,m,s. Example: '2d-ago' will get everything that is up to 2 days old. Can also send time in ms since epoch or as a datetime object.
- **end** (*str, int, datetime*) – Get datapoints up to this time. The format is the same as for start.

class `cognite.client.stable.datapoints.DatapointsResponse` (*internal_representation*)

Bases: `cognite.client._api_client.CogniteResponse`

Datapoints Response Object.

next_cursor ()

Returns next cursor to use for paging through results. Returns `None` if there are no more results.

previous_cursor ()

Returns previous cursor to use for paging through results. Returns `None` if there are no more results.

to_json ()

Returns data as a json object

to_pandas()
Returns data as a pandas dataframe

class `cognite.client.stable.datapoints.DatapointsResponseIterator` (*datapoints_objects*)
Bases: `object`

Iterator for Datapoints Response Objects.

class `cognite.client.stable.datapoints.LatestDatapointResponse` (*internal_representation*)
Bases: `cognite.client._api_client.CogniteResponse`

Latest Datapoint Response Object.

next_cursor()
Returns next cursor to use for paging through results. Returns `None` if there are no more results.

previous_cursor()
Returns previous cursor to use for paging through results. Returns `None` if there are no more results.

to_json()
Returns data as a json object

to_pandas()
Returns data as a pandas dataframe

class `cognite.client.stable.datapoints.TimeseriesWithDatapoints` (*name*, *datapoints*)
Bases: `object`

Data transfer object for a timeseries with datapoints.

Parameters

- **name** (*str*) – Unique ID of time series.
- **datapoints** (*List [stable.datapoints.Datapoint]*) – List of datapoints in the timeseries.

2.3 Events

2.3.1 Client

class `cognite.client.stable.events.EventsClient` (***kwargs*)

delete_events (*event_ids: List[int]*) → `None`
Deletes a list of events.

Parameters **event_ids** (*List [int]*) – List of ids of events to delete.

Returns `None`

Examples

Deleting a list of events:

```
client = CogniteClient()
res = client.events.delete_events(event_ids=[1, 2, 3, 4, 5])
```

get_event (*event_id: int*) → `cognite.client.stable.events.EventResponse`
Returns a `EventResponse` containing an event matching the id.

Parameters **event_id** (*int*) – The event id.

Returns A data object containing the requested event.

Return type `stable.events.EventResponse`

Examples

Getting an event:

```
client = CogniteClient()
res = client.events.get_event(123)
print(res)
```

get_events (*type=None, sub_type=None, asset_id=None, **kwargs*) → `cognite.client.stable.events.EventListResponse`
Returns an `EventListResponse` object containing events matching the query.

Parameters

- **type** (*str*) – Type (class) of event, e.g. ‘failure’.
- **sub_type** (*str*) – Sub-type of event, e.g. ‘electrical’.
- **asset_id** (*int*) – Return events associated with this assetId.

Keyword Arguments

- **sort** (*str*) – Sort descending or ascending. Default ‘ASC’.
- **cursor** (*str*) – Cursor to use for paging through results.
- **limit** (*int*) – Return up to this many results. Maximum is 10000. Default is 25.
- **has_description** (*bool*) – Return only events that have a textual description. Default null. False gives only those without description.
- **min_start_time** (*string*) – Only return events from after this time.
- **max_start_time** (*string*) – Only return events form before this time.
- **autopaging** (*bool*) – Whether or not to automatically page through results. If set to true, limit will be disregarded. Defaults to False.

Returns A data object containing the requested event.

Return type `stable.events.EventListResponse`

Examples

Getting all events of a given type:

```
client = CogniteClient()
res = client.events.get_events(type="a special type", autopaging=True)
print(res.to_pandas())
```

post_events (*events: List[cognite.client.stable.events.Event]*) → `cognite.client.stable.events.EventListResponse`
Adds a list of events and returns an `EventListResponse` object containing created events.

Parameters `events` (`List[stable.events.Event]`) – List of events to create.

Returns `stable.events.EventListResponse`

Examples

Posting two events and linking them to an asset:

```
from cognite.client.stable.events import Event
client = CogniteClient()

my_events = [Event(start_time=1, end_time=10, type="workorder", asset_
↳ids=[123]),
              Event(start_time=11, end_time=20, type="workorder", asset_
↳ids=[123])]
res = client.events.post_events(my_events)
print(res)
```

search_for_events (`description=None`, `type=None`, `subtype=None`, `min_start_time=None`, `max_start_time=None`, `min_end_time=None`, `max_end_time=None`, `min_created_time=None`, `max_created_time=None`, `min_last_updated_time=None`, `max_last_updated_time=None`, `metadata=None`, `asset_ids=None`, `asset_subtrees=None`, `**kwargs`)

Search for events.

Parameters

- **description** (`str`) – Prefix and fuzzy search on description.
- **type** (`str`) – Filter on type (case-sensitive).
- **subtype** (`str`) – Filter on subtype (case-sensitive).
- **min_start_time** (`str`) – Filter out events with `startTime` before this. Format is milliseconds since epoch.
- **max_start_time** (`str`) – Filter out events with `startTime` after this. Format is milliseconds since epoch.
- **min_end_time** (`str`) – Filter out events with `endTime` before this. Format is milliseconds since epoch.
- **max_end_time** (`str`) – Filter out events with `endTime` after this. Format is milliseconds since epoch.
- **min_created_time** (`str`) – Filter out events with `createdTime` before this. Format is milliseconds since epoch.
- **max_created_time** (`str`) – Filter out events with `createdTime` after this. Format is milliseconds since epoch.
- **min_last_updated_time** (`str`) – Filter out events with `lastUpdatedTime` before this. Format is milliseconds since epoch.
- **max_last_updated_time** (`str`) – Filter out events with `lastUpdatedTime` after this. Format is milliseconds since epoch.
- **metadata** (`dict`) – Filter out events that do not match these metadata fields and values (case-sensitive). Format is `{"key1": "value1", "key2": "value2"}`.
- **asset_ids** (`List[int]`) – Filter out events that are not linked to any of these assets. Format is `[12,345,6,7890]`.

- **asset_subtrees** (*List [int]*) – Filter out events that are not linked to assets in the subtree rooted at these assets. Format is [12,345,6,7890].

Keyword Arguments

- **sort** (*str*) – Field to be sorted.
- **dir** (*str*) – Sort direction (desc or asc)
- **limit** (*int*) – Return up to this many results. Max is 1000, default is 25.
- **offset** (*int*) – Offset from the first result. Sum of limit and offset must not exceed 1000. Default is 0.

Returns stable.events.EventListResponse.

Examples

Perform a fuzzy search on event descriptions and get the first 3 results:

```
client = CogniteClient()
res = client.events.search_for_events(description="Something like this",
↪ limit=10)
print(res)
```

2.3.2 Data Objects

class cognite.client.stable.events.**Event** (*start_time=None, end_time=None, description=None, type=None, sub_type=None, metadata=None, asset_ids=None*)

Bases: object

Data transfer object for events.

Parameters

- **start_time** (*int*) – Start time of the event in ms since epoch.
- **end_time** (*int*) – End time of the event in ms since epoch.
- **description** (*str*) – Textual description of the event.
- **type** (*str*) – Type of the event, e.g. ‘failure’.
- **sub_type** (*str*) – Subtype of the event, e.g. ‘electrical’.
- **metadata** (*dict*) – Customizable extra data about the event.
- **asset_ids** (*list [int]*) – List of Asset IDs of related equipments that this event relates to.

class cognite.client.stable.events.**EventListResponse** (*internal_representation*)

Bases: *cognite.client._api_client.CogniteCollectionResponse*

Event List Response Object.

next_cursor ()

Returns next cursor to use for paging through results. Returns None if there are no more results.

previous_cursor ()

Returns previous cursor to use for paging through results. Returns None if there are no more results.

`to_json()`
Returns data as a json object

`to_pandas()`

class `cognite.client.stable.events.EventResponse` (*internal_representation*)
Bases: `cognite.client._api_client.CogniteResponse`

Event Response Object.

`next_cursor()`
Returns next cursor to use for paging through results. Returns `None` if there are no more results.

`previous_cursor()`
Returns previous cursor to use for paging through results. Returns `None` if there are no more results.

`to_json()`
Returns data as a json object

`to_pandas()`

2.4 Files

2.4.1 Client

class `cognite.client.stable.files.FilesClient` (**kwargs)

`delete_files` (*file_ids*) → List
Delete

Parameters `file_ids` (*list[int]*) – List of IDs of files to delete.

Returns List of files deleted and files that failed to delete.

Examples

Delete two files:

```
client = CogniteClient()
res = client.files.delete_files([123, 234])
```

`download_file` (*id: int, get_contents: bool = False*) → Union[str, bytes]
Get list of files matching query.

Parameters

- `id` (*int*) – Path to file to upload, if omitted a upload link will be returned.
- `get_contents` (*bool, optional*) – Boolean to determine whether or not to return file contents as string. Default is `False` and download url is returned.

Returns Download link if `get_contents` is `False` else file contents.

Return type Union[str, bytes]

Examples

Get a download url for the file:

```
client = CogniteClient()
res = client.files.download_file(id=12345)
download_url = res["downloadUrl"]
```

Download a file:

```
client = CogniteClient()
file_bytes = client.files.download_file(id=12345, get_contents=True)
```

get_file_info (*id*) → cognite.client.stable.files.FileInfoResponse
Returns information about a file.

Parameters *id* (*int*) – Id of the file.

Returns A data object containing the requested file information.

Return type *stable.files.FileInfoResponse*

Examples

Get info about a specific file:

```
client = CogniteClient()
res = client.files.get_file_info(12345)
print(res)
```

list_files (*name=None, directory=None, file_type=None, source=None, **kwargs*) → cognite.client.stable.files.FileListResponse
Get list of files matching query.

Parameters

- **name** (*str, optional*) – List all files with this name.
- **directory** (*str, optional*) – Directory to list files from.
- **source** (*str, optional*) – List files coming from this source.
- **file_type** (*str, optional*) – Type of files to list.

Keyword Arguments

- **asset_id** (*list*) – Returns all files associated with this asset id.
- **sort** (*str*) – Sort descending or ascending. ‘ASC’ or ‘DESC’.
- **limit** (*int*) – Number of results to return.
- **is_uploaded** (*bool*) – List only uploaded files if true. If false, list only other files. If not set, list all files without considering whether they are uploaded or not.
- **autopaging** (*bool*) – Whether or not to automatically page through results. If set to true, limit will be disregarded. Defaults to False.
- **cursor** (*str*) – Cursor to use for paging through results.

Returns A data object containing the requested files information.

Return type *stable.files.FileListResponse*

Examples

List all files in a given directory:

```
client = CogniteClient()
res = client.files.list_files(directory="allfiles/myspecialfiles",
↪ autopaging=True)
print(res.to_pandas())
```

upload_file (*file_name*, *file_path=None*, *directory=None*, *source=None*, *file_type=None*, *content_type=None*, ***kwargs*) → Dict
Upload metadata about a file and get an upload link.

The link will expire after 30 seconds if not resumable. A resumable upload link is default. Such a link is one-time use and expires after one week. For more information, check this link: https://cloud.google.com/storage/docs/json_api/v1/how-tos/resumable-upload. Use PUT request to upload file with the link returned.

If *file_path* is specified, the file will be uploaded directly by the SDK.

Parameters

- **file_name** (*str*) – File name. Max length is 256.
- **file_path** (*str*, *optional*) – Path of file to upload, if omitted a upload link will be returned.
- **content_type** (*str*, *optional*) – MIME type of your file. Required if *file_path* is specified.
- **directory** (*str*, *optional*) – Directory containing the file. Max length is 512.
- **source** (*str*, *optional*) – Source that this file comes from. Max length is 256.
- **file_type** (*str*, *optional*) – File type. E.g. pdf, css, spreadsheet, .. Max length is 64.

Keyword Arguments

- **metadata** (*dict*) – Customized data about the file.
- **asset_ids** (*list*) – IDs of assets related to this file.
- **resumable** (*bool*) – Whether to generate a resumable URL or not. Default is true.
- **overwrite** (*bool*) – Whether to overwrite existing data if duplicate or not. Default is false.

Returns A dictionary containing the field *fileId* and optionally also *uploadURL* if *file_path* is omitted.

Return type Dict

Examples

Upload a file and link it to an asset:

```
client = CogniteClient()
res = client.files.upload_file(file_name="myfile", file_path="/path/to/my/
↪ file.txt",
    content_type="text/plain", asset_ids=[123])
file_id = res["fileId"]
```

2.4.2 Data Objects

class `cognite.client.stable.files.FileInfoResponse` (*internal_representation*)

Bases: `cognite.client._api_client.CogniteResponse`

File Info Response Object.

Parameters

- **id** (*int*) – ID given by the API to the file.
- **file_name** (*str*) – File name. Max length is 256.
- **directory** (*str*) – Directory containing the file. Max length is 512.
- **source** (*dict*) – Source that this file comes from. Max length is 256.
- **file_type** (*str*) – File type. E.g. pdf, css, spreadsheet, .. Max length is 64.
- **metadata** (*dict*) – Customized data about the file.
- **asset_ids** (*list[str]*) – Names of assets related to this file.
- **uploaded** (*bool*) – Whether or not the file is uploaded.
- **uploaded_at** (*int*) – Epoch time (ms) when the file was uploaded successfully.

next_cursor ()

Returns next cursor to use for paging through results. Returns `None` if there are no more results.

previous_cursor ()

Returns previous cursor to use for paging through results. Returns `None` if there are no more results.

to_json ()

Returns data as a json object

to_pandas ()

class `cognite.client.stable.files.FileListResponse` (*internal_representation*)

Bases: `cognite.client._api_client.CogniteCollectionResponse`

File List Response Object

next_cursor ()

Returns next cursor to use for paging through results. Returns `None` if there are no more results.

previous_cursor ()

Returns previous cursor to use for paging through results. Returns `None` if there are no more results.

to_json ()

Returns data as a json object

to_pandas ()

2.5 Login

2.5.1 Client

```
class cognite.client.stable.login.LoginClient (request_session: requests.sessions.Session, version: str = None, project: str = None, base_url: str = None, num_of_workers: int = None, cookies: Dict = None, headers: Dict = None, timeout: int = None)
```

status () → cognite.client.stable.login.LoginStatusResponse

Check login status

Returns client.stable.login.LoginStatusResponse

Examples

Check the current login status:

```
client = CogniteClient()
login_status = client.login.status()
print(login_status)
```

2.5.2 Data Objects

```
class cognite.client.stable.login.LoginStatusResponse (internal_representation)
  Bases: cognite.client._api_client.CogniteResponse
```

user

str – Current user

logged_in

bool – Is user logged in

project

str – Current project

project_id

str – Current project id

next_cursor ()

Returns next cursor to use for paging through results. Returns `None` if there are no more results.

previous_cursor ()

Returns previous cursor to use for paging through results. Returns `None` if there are no more results.

to_json ()

Returns data as a json object

2.6 Raw

2.6.1 Client

```
class cognite.client.stable.raw.RawClient (**kwargs)
```

create_databases (*database_names: list*) → *cognite.client.stable.raw.RawResponse*
Creates databases in the Raw API and returns the created databases.

Parameters **database_names** (*list*) – A list of databases to create.

Returns A data object containing the requested data with several getter methods with different output formats.

Return type *stable.raw.RawResponse*

create_rows (*database_name: str = None, table_name: str = None, rows: List[cognite.client.stable.raw.RawRow] = None, ensure_parent=False, use_gzip=True*) → *None*
Creates tables in the given Raw API database.

Parameters

- **database_name** (*str*) – The database to create rows in.
- **table_name** (*str*) – The table names to create rows in.
- **rows** (*list[stable.raw.RawRow]*) – The rows to create.
- **ensure_parent** (*bool*) – Create database/table if it doesn't exist already
- **use_gzip** (*bool*) – Compress content using gzip

Returns *None*

create_tables (*database_name: str = None, table_names: list = None*) → *cognite.client.stable.raw.RawResponse*
Creates tables in the given Raw API database.

Parameters

- **database_name** (*str*) – The database to create tables in.
- **table_names** (*list*) – The table names to create.

Returns A data object containing the requested data with several getter methods with different output formats.

Return type *stable.raw.RawResponse*

delete_databases (*database_names: list, recursive: bool = False*) → *None*
Deletes databases in the Raw API.

Parameters **database_names** (*list*) – A list of databases to delete.

Returns *None*

delete_rows (*database_name: str = None, table_name: str = None, rows: List[cognite.client.stable.raw.RawRow] = None*) → *None*
Deletes rows in the Raw API.

Parameters

- **database_name** (*str*) – The database to create tables in.

- **table_name** (*str*) – The table name where the rows are at.
- **rows** (*list*) – The rows to delete.

Returns None

delete_tables (*database_name: str = None, table_names: list = None*) → None
Deletes databases in the Raw API.

Parameters

- **database_name** (*str*) – The database to create tables in.
- **table_names** (*list*) – The table names to create.

Returns None

get_databases (*limit: int = None, cursor: str = None*) → `cognite.client.stable.raw.RawResponse`
Returns a RawObject containing a list of raw databases.

Parameters

- **limit** (*int*) – A limit on the amount of results to return.
- **cursor** (*str*) – A cursor can be provided to navigate through pages of results.

Returns A data object containing the requested data with several getter methods with different output formats.

Return type `stable.raw.RawResponse`

get_row (*database_name: str = None, table_name: str = None, row_key: str = None*) → `cognite.client.stable.raw.RawResponse`
Returns a RawObject containing a list of rows.

Parameters

- **database_name** (*str*) – The database name to retrieve rows from.
- **table_name** (*str*) – The table name to retrieve rows from.
- **row_key** (*str*) – The key of the row to fetch.

Returns A data object containing the requested data with several getter methods with different output formats.

Return type `stable.raw.RawResponse`

get_rows (*database_name: str = None, table_name: str = None, limit: int = None, cursor: str = None*) → `cognite.client.stable.raw.RawResponse`
Returns a RawObject containing a list of rows.

Parameters

- **database_name** (*str*) – The database name to retrieve rows from.
- **table_name** (*str*) – The table name to retrieve rows from.
- **limit** (*int*) – A limit on the amount of results to return.
- **cursor** (*str*) – A cursor can be provided to navigate through pages of results.

Returns A data object containing the requested data with several getter methods with different output formats.

Return type `stable.raw.RawResponse`

`get_tables` (*database_name*: *str* = *None*, *limit*: *int* = *None*, *cursor*: *str* = *None*) → `cognite.client.stable.raw.RawResponse`
Returns a `RawObject` containing a list of tables in a raw database.

Parameters

- **database_name** (*str*) – The database name to retrieve tables from.
- **limit** (*int*) – A limit on the amount of results to return.
- **cursor** (*str*) – A cursor can be provided to navigate through pages of results.

Returns A data object containing the requested data with several getter methods with different output formats.

Return type `stable.raw.RawResponse`

2.6.2 Data Objects

class `cognite.client.stable.raw.RawResponse` (*internal_representation*)

Bases: `cognite.client._api_client.CogniteResponse`

Raw Response Object.

next_cursor ()

Returns next cursor to use for paging through results. Returns `None` if there are no more results.

previous_cursor ()

Returns previous cursor to use for paging through results. Returns `None` if there are no more results.

to_json ()

Returns data as a json object

to_pandas ()

Returns data as a pandas dataframe

class `cognite.client.stable.raw.RawRow` (*key*, *columns*)

Bases: `object`

DTO for a row in a raw database.

The Raw API is a simple key/value-store. Each row in a table in a raw database consists of a unique row key and a set of columns.

Parameters

- **key** (*str*) – Unique key for the row.
- **columns** (*dict*) – A key/value-map consisting of the values in the row.

repr_json ()

2.7 Tagmatching

2.7.1 Client

class `cognite.client.stable.tagmatching.TagMatchingClient` (***kwargs*)

tag_matching (*tag_ids*, *fuzzy_threshold=0*, *platform=None*) → `cognite.client.stable.tagmatching.TagMatchingResponse`

Returns a `TagMatchingObject` containing a list of matched tags for the given query.

This method takes an arbitrary string as argument and performs fuzzy matching with a user defined threshold toward tag ids in the system.

Parameters

- **tag_ids** (*list*) – The tag_ids to retrieve matches for.
- **fuzzy_threshold** (*int*) – The threshold to use when searching for matches. A fuzzy threshold of 0 means you only want to accept perfect matches. Must be ≥ 0 .
- **platform** (*str*) – The platform to search on.

Returns A data object containing the requested data with several getter methods with different output formats.

Return type `stable.tagmatching.TagMatchingResponse`

2.7.2 Data Objects

class `cognite.client.stable.tagmatching.TagMatchingResponse` (*internal_representation*)

Bases: `cognite.client._api_client.CogniteResponse`

Tag Matching Response Object.

In addition to the standard output formats this data object also has a `to_list()` method which returns a list of names of the tag matches.

next_cursor ()

Returns next cursor to use for paging through results. Returns `None` if there are no more results.

previous_cursor ()

Returns previous cursor to use for paging through results. Returns `None` if there are no more results.

to_json ()

Returns data as a json object

to_list (*first_matches_only=True*)

Returns a list representation of the matches.

Parameters **first_matches_only** (*bool*) – Boolean determining whether or not to return only the top match for each tag.

Returns list of matched tags.

Return type list

to_pandas ()

Returns data as a pandas dataframe

2.8 Time Series

2.8.1 Client

class `cognite.client.stable.time_series.TimeSeriesClient` (***kwargs*)

delete_time_series (*name*) → None

Delete a timeseries.

Parameters **name** (*str*) – Name of timeseries to delete.

Returns None

Examples

Delete a time series by name:

```
client = CogniteClient()

client.time_series.delete_time_series(name="my_ts_1")
```

get_time_series (*prefix=None*, *description=None*, *include_metadata=False*,
asset_id=None, *path=None*, ***kwargs*) → `cognite.client.stable.time_series.TimeSeriesListResponse`

Returns an object containing the requested timeseries.

Parameters

- **prefix** (*str*) – List timeseries with this prefix in the name.
- **description** (*str*) – Filter timeseries that contains this string in its description.
- **include_metadata** (*bool*) – Decide if the metadata field should be returned or not. Defaults to False.
- **asset_id** (*int*) – Get timeseries related to this asset.
- **path** (*str*) – Get timeseries under this asset path branch.

Keyword Arguments

- **limit** (*int*) – Number of results to return.
- **autopaging** (*bool*) – Whether or not to automatically page through results. If set to true, limit will be disregarded. Defaults to False.

Returns A data object containing the requested timeseries with several getter methods with different output formats.

Return type `stable.time_series.TimeSeriesListResponse`

Examples

Get all time series for a given asset:

```
client = CogniteClient()
res = client.time_series.get_time_series(asset_id=123, autopaging=True)
print(res.to_pandas())
```

post_time_series (*time_series: List[cognite.client.stable.time_series.TimeSeries]*) → None

Create a new time series.

Parameters **time_series** (*list[stable.time_series.TimeSeries]*) – List of time series data transfer objects to create.

Returns None

Examples

Create a new time series:

```
from cognite.client.stable.time_series import TimeSeries
client = CogniteClient()

my_time_series = [TimeSeries(name="my_ts_1")]

client.time_series.post_time_series(my_time_series)
```

update_time_series (*time_series: List[cognite.client.stable.time_series.TimeSeries]*) → None

Update an existing time series.

For each field that can be updated, a null value indicates that nothing should be done.

Parameters **time_series** (*list[stable.time_series.TimeSeries]*) – List of time series data transfer objects to update.

Returns None

Examples

Update the unit of a time series:

```
from cognite.client.stable.time_series import TimeSeries
client = CogniteClient()

my_time_series = [TimeSeries(name="my_ts_1", unit="celsius")]

client.time_series.update_time_series(my_time_series)
```

2.8.2 Data Objects

class `cognite.client.stable.time_series.TimeSeries` (*name, is_string=False, metadata=None, unit=None, asset_id=None, description=None, security_categories=None, is_step=None*)

Bases: `object`

Data Transfer Object for a time series.

Parameters

- **name** (*str*) – Unique name of time series.
- **is_string** (*bool*) – Whether the time series is string valued or not.
- **metadata** (*dict*) – Metadata.
- **unit** (*str*) – Physical unit of the time series.
- **asset_id** (*int*) – Asset that this time series belongs to.
- **description** (*str*) – Description of the time series.
- **security_categories** (*list(int)*) – Security categories required in order to access this time series.

- **is_step** (*bool*) – Whether or not the time series is a step series.

class `cognite.client.stable.time_series.TimeSeriesListResponse` (*internal_representation*)

Bases: `cognite.client._api_client.CogniteCollectionResponse`

Time series Response Object

next_cursor ()

Returns next cursor to use for paging through results. Returns `None` if there are no more results.

previous_cursor ()

Returns previous cursor to use for paging through results. Returns `None` if there are no more results.

to_json ()

Returns data as a json object

to_pandas (*include_metadata: bool = False*)

Returns data as a pandas dataframe

Parameters **include_metadata** (*bool*) – Whether or not to include metadata fields in the resulting dataframe

class `cognite.client.stable.time_series.TimeSeriesResponse` (*internal_representation*)

Bases: `cognite.client._api_client.CogniteResponse`

Time series Response Object

next_cursor ()

Returns next cursor to use for paging through results. Returns `None` if there are no more results.

previous_cursor ()

Returns previous cursor to use for paging through results. Returns `None` if there are no more results.

to_json ()

Returns data as a json object

to_pandas ()

Returns data as a pandas dataframe

3.1 Analytics

3.1.1 Models

Client

```
class cognite.client.experimental.analytics.models.ModelsClient (**kwargs)
```

```
create_model (name: str, description: str = "", metadata: Dict[str, Any] = None, input_fields: List[str] = None, output_fields: List[str] = None) → Dict
```

Creates a new model

Parameters

- **name** (*str*) – Name of model
- **description** (*str*) – Description
- **metadata** (*Dict[str, Any]*) – Metadata about model
- **input_fields** (*List[str]*) – List of input fields the model accepts
- **(List[str]) (output_fields)** – List of output fields the model produces

Returns The created model.

Return type Dict

```
create_schedule (model_id: int, name: str, output_data_spec: Dict, input_data_spec: Dict, description: str = None, args: Dict = None, metadata: Dict = None) → Dict
```

Create a new schedule on a given model.

Parameters

- **model_id** (*int*) – Id of model to create schedule on
- **name** (*str*) – Name of schedule

- **output_data_spec** (*Dict*) – Specification of output. Example below.
- **input_data_spec** (*Dict*) – Specification of input. Example below.
- **description** (*str*) – Description for schedule
- **args** (*Dict*) – Dictionary of keyword arguments to pass to predict method.
- **metadata** (*Dict*) – Dictionary of metadata about schedule

Returns The created schedule.

Return type Dict

Examples The output data spec must look like this:

```
{
  "timeSeries": [
    {
      "label": "string",
      "id": 123456789
    }
  ]
}
```

The input data spec must look like this. The local aggregate and the missingDataStrategy fields are optional:

```
{
  "windowSize": "1s",
  "stride": "1s",
  "missingDataStrategy": "",
  "timeSeries": [
    {
      "label": "string",
      "id": 0,
      "missingDataStrategy": "string",
      "aggregate": "string"
    }
  ],
  "aggregate": "string",
  "granularity": "string"
}
```

create_source_package (*name: str, package_name: str, available_operations: List[str], runtime_version: str, description: str = None, meta_data: Dict = None, file_path: str = None*) → Dict

Upload a source package to the model hosting environment.

Parameters

- **name** (*str*) – Name of source package
- **package_name** (*str*) – name of root package for model
- **available_operations** (*List[str]*) – List of routines which this source package supports [“predict”, “train”]
- **runtime_version** (*str*) – Version of environment in which the source-package should run. Currently only 0.1.
- **description** (*str*) – Description for source package

- **meta_data** (*Dict*) – User defined key value pair of additional information.
- **file_path** (*str*) – File path of source package distribution. If not specified, a download url will be returned.

Returns Source package ID if file path was specified. Else, source package id and upload url.

Return type Dict

delete_model (*model_id: int*) → None

Delete a model.

Will also delete all versions and schedules for this model.

Parameters **model_id** (*int*) – Delete model with this id.

Returns None

delete_schedule (*schedule_id: int*) → None

Delete a schedule by id.

Parameters **schedule_id** (*int*) – The id of the schedule to delete.

Returns None

delete_source_package (*source_package_id: int*) → None

Delete source package by id.

Parameters **source_package_id** (*int*) – Id of source package to delete.

Returns None

delete_version (*model_id: int, version_id: int*) → None

Delete a model version by id.

Parameters

- **model_id** (*int*) – Id of model which has the model version.
- **version_id** (*int*) – Id of model version.

Returns None

get_model (*model_id: int*) → Dict

Get a model by id.

Parameters **model_id** (*int*) – Id of model to get.

Returns The requested model

Return type Dict

get_models () → List[Dict]

Get all models.

Returns List of models

Return type List[Dict]

get_schedule (*schedule_id: int*) → Dict

Get a schedule by id.

Returns The requested schedule.

Return type Dict

get_schedules () → List[Dict]

Get all schedules.

Returns The requested schedules.

Return type List[Dict]

get_source_package (*source_package_id: int*) → Dict

Get model source package by id.

Parameters **source_package_id** (*int*) – Id of source package to get.

Returns The requested source package.

Return type Dict

get_source_packages () → List[Dict]

Get all model source packages.

Returns List of source packages.

Return type List[Dict]

get_version (*model_id: int, version_id: int*) → Dict

Get a specific model version by id.

Parameters

- **model_id** (*int*) – Id of model which has the model version.
- **version_id** (*int*) – Id of model version.

Returns The requested model version

Return type Dict

get_versions (*model_id: int*) → List[Dict]

Get all versions of a specific model.

Parameters **model_id** (*int*) – Get versions for the model with this id.

Returns List of model versions

Return type List[Dict]

online_predict (*model_id: int, version_id: int = None, instances: List = None, args: Dict[str, Any] = None*) → List

Perform online prediction on a models active version or a specified version.

Parameters

- **model_id** (*int*) – Perform a prediction on the model with this id. Will use active version.
- **version_id** (*int*) – Use this version instead of the active version. (optional)
- **instances** (*List*) – List of JSON serializable instances to pass to your model one-by-one.
- **args** (*Dict[str, Any]*) –

Returns List of predictions for each instance.

Return type List

train_model_version (*model_id: int, name: str, source_package_id: int, train_source_package_id: int = None, metadata: Dict = None, description: str = None, args: Dict[str, Any] = None, scale_tier: str = None, machine_type: str = None*) → Dict

Train a new version of a model.

Parameters

- **model_id** (*int*) – Create a new version under the model with this id
- **name** (*str*) – Name of model version. Must be unique on the model.
- **source_package_id** (*int*) – Use the source package with this id
- **train_source_package_id** (*int*) – Use this source package for training. If omitted, will default to `source_package_id`.
- **metadata** (*Dict[str, Any]*) – Metadata about model version
- **description** (*str*) – Description of model version
- **args** (*Dict[str, Any]*) – Dictionary of arguments to pass to the training job.
- **scale_tier** (*str*) – Which scale tier to use. Must be either “BASIC” or “CUSTOM”
- **machine_type** (*str*) – Specify a machine type Applies only if `scale_tier` is “CUSTOM”.

Returns The created model version.

Return type Dict

Data Objects**3.2 Time Series****3.2.1 Client**

```
class cognite.client.experimental.time_series.TimeSeriesClient (**kwargs)
```

delete_time_series_by_id (*ids: List[int]*) → None

Delete multiple time series by id.

Parameters **ids** (*List[int]*) – IDs of time series to delete.

Returns None

Examples

Delete a single time series by id:

```
client = CogniteClient()
client.time_series.delete_time_series_by_id(ids=[my_ts_id])
```

get_multiple_time_series_by_id (*ids: List[int]*) → `cognite.client.experimental.time_series.TimeSeriesResponse`

Returns a `TimeseriesResponse` object containing the requested timeseries.

Parameters **ids** (*List[int]*) – IDs of timeseries to look up

Returns A data object containing the requested timeseries with several getter methods with different output formats.

Return type `client.experimental.time_series.TimeSeriesResponse`

get_time_series_by_id (*id*: *int*) → *cognite.client.experimental.time_series.TimeSeriesResponse*
Returns a *TimeseriesResponse* object containing the requested timeseries.

Parameters *id* (*int*) – ID of timeseries to look up

Returns A data object containing the requested timeseries.

Return type *client.experimental.time_series.TimeSeriesResponse*

search_for_time_series (*name=None*, *description=None*, *query=None*, *unit=None*,
is_string=None, *is_step=None*, *metadata=None*, *asset_ids=None*, *asset_subtrees=None*, *min_created_time=None*,
max_created_time=None, *min_last_updated_time=None*, *max_last_updated_time=None*, ***kwargs*) → *cog-*
nite.client.experimental.time_series.TimeSeriesResponse

Returns a *TimeSeriesResponse* object containing the search results.

Parameters

- **name** (*str*) – Prefix and fuzzy search on name.
- **description** (*str*) – Prefix and fuzzy search on description.
- **query** (*str*) – Search on name and description using wildcard search on each of the words (separated by spaces). Retrieves results where at least on word must match. Example: “some other”
- **unit** (*str*) – Filter on unit (case-sensitive)
- **is_string** (*bool*) – Filter on whether the ts is a string ts or not.
- **is_step** (*bool*) – Filter on whether the ts is a step ts or not.
- **metadata** (*Dict*) – Filter out time series that do not match these metadata fields and values (case-sensitive). Format is {“key1”: “val1”, “key2”, “val2”}
- **asset_ids** (*List*) – Filter out time series that are not linked to any of these assets. Format is [12,345,6,7890].
- **asset_subtrees** (*List*) – Filter out time series that are not linked to assets in the subtree rooted at these assets. Format is [12,345,6,7890].
- **min_created_time** (*int*) – Filter out time series with *createdTime* before this. Format is milliseconds since epoch.
- **max_created_time** (*int*) – Filter out time series with *createdTime* after this. Format is milliseconds since epoch.
- **min_last_updated_time** (*int*) – Filter out time series with *lastUpdatedTime* before this. Format is milliseconds since epoch.
- **max_last_updated_time** (*int*) – Filter out time series with *lastUpdatedTime* after this. Format is milliseconds since epoch.

Keyword Arguments

- **sort** (*str*) – “*createdTime*” or “*lastUpdatedTime*”. Field to be sorted. If not specified, results are sorted by relevance score.
- **dir** (*str*) – “*asc*” or “*desc*”. Only applicable if *sort* is specified. Default ‘*desc*’.
- **limit** (*int*) – Return up to this many results. Maximum is 1000. Default is 25.
- **offset** (*int*) – Offset from the first result. Sum of *limit* and *offset* must not exceed 1000. Default is 0.

- **boost_name** (*bool*) – Whether or not boosting name field. This option is `test_experimental` and can be changed.

Returns A data object containing the requested timeseries with several getter methods with different output formats.

Return type `client.experimental.time_series.TimeSeriesResponse`

3.2.2 Data Objects

class `cognite.client.experimental.time_series.TimeSeriesResponse` (*internal_representation*)

Bases: `cognite.client._api_client.CogniteResponse`

Time series Response Object

next_cursor ()

Returns next cursor to use for paging through results. Returns `None` if there are no more results.

previous_cursor ()

Returns previous cursor to use for paging through results. Returns `None` if there are no more results.

to_json ()

Returns data as a json object

to_pandas (*include_metadata: bool = False*)

Returns data as a pandas dataframe

Parameters **include_metadata** (*bool*) – Whether or not to include metadata fields in the resulting dataframe

3.3 Datapoints

3.3.1 Client

class `cognite.client.experimental.datapoints.DatapointsClient` (***kwargs*)

get_datapoints (*id, start, end=None, aggregates=None, granularity=None, **kwargs*) → `cognite.client.experimental.datapoints.DatapointsResponse`

Returns a `DatapointsObject` containing a list of datapoints for the given query.

This method will automate paging for the user and return all data for the given time period.

Parameters

- **id** (*int*) – The unique id of the timeseries to retrieve data for.
- **start** (*Union[str, int, datetime]*) – Get datapoints after this time. Format is `N[timeunit]-ago` where `timeunit` is `w,d,h,m,s`. E.g. `'2d-ago'` will get everything that is up to 2 days old. Can also send time in `ms` since epoch or a `datetime` object which will be converted to `ms` since epoch UTC.
- **end** (*Union[str, int, datetime]*) – Get datapoints up to this time. Same format as for `start`.
- **aggregates** (*list*) – The list of aggregate functions you wish to apply to the data. Valid aggregate functions are: `'average/avg, max, min, count, sum, interpolation/int, stepinterpolation/step'`.

- **granularity** (*str*) – The granularity of the aggregate values. Valid entries are : ‘day/d, hour/h, minute/m, second/s’, or a multiple of these indicated by a number as a prefix e.g. ‘12hour’.

Keyword Arguments

- **include_outside_points** (*bool*) – No description
- **workers** (*int*) – Number of download processes to run in parallel. Defaults to number returned by `cpu_count()`.
- **limit** (*str*) – Max number of datapoints to return. If limit is specified, this method will not automate paging and will return a maximum of 100,000 dps.

Returns A data object containing the requested data with several getter methods with different output formats.

Return type `client.test_experimental.datapoints.DatapointsResponse`

3.3.2 Data Objects

class `cognite.client.experimental.datapoints.DatapointsResponse` (*internal_representation*)

Bases: `cognite.client._api_client.CogniteResponse`

Datapoints Response Object.

next_cursor ()

Returns next cursor to use for paging through results. Returns `None` if there are no more results.

previous_cursor ()

Returns previous cursor to use for paging through results. Returns `None` if there are no more results.

to_json ()

Returns data as a json object

to_pandas ()

Returns data as a pandas dataframe

3.4 Sequences

3.4.1 Client

class `cognite.client.experimental.sequences.SequencesClient` (***kwargs*)

delete_sequence_by_id (*id: int*) → `None`

Deletes the sequence with the given id.

Parameters *id* (*int*) – ID of the sequence to delete

Returns `None`

get_data_from_sequence (*id: int, inclusive_from: int = None, inclusive_to: int = None, limit: int = 100, column_ids: List[int] = None*) → `cognite.client.experimental.sequences.SequenceDataResponse`

Gets data from the given sequence.

Parameters

- **id** (*int*) – id of the sequence.

- **inclusive_from** (*int*) – Row number to get from (inclusive). If set to None, you'll get data from the first row that exists.
- **inclusive_to** (*int*) – Row number to get to (inclusive). If set to None, you'll get data to the last row that exists (depending on the limit).
- **limit** (*int*) – How many rows to return.
- **column_ids** (*List[int]*) – ids of the columns to get data for.

Returns A data object containing the requested sequence.

Return type `client.test_experimental.sequences.SequenceDataResponse`

get_sequence_by_external_id (*external_id: str*) → `cognite.client.experimental.sequences.Sequence`

Returns a Sequence object containing the requested sequence.

Parameters **external_id** (*int*) – External ID of the sequence to look up

Returns A data object containing the requested sequence.

Return type `test_experimental.dto.Sequence`

get_sequence_by_id (*id: int*) → `cognite.client.experimental.sequences.Sequence`

Returns a Sequence object containing the requested sequence.

Parameters **id** (*int*) – ID of the sequence to look up

Returns A data object containing the requested sequence.

Return type `client.test_experimental.sequences.Sequence`

post_data_to_sequence (*id: int, rows: List[cognite.client.experimental.sequences.Row]*) → None

Posts data to a sequence.

Parameters

- **id** (*int*) – ID of the sequence.
- **rows** (*list*) – List of rows with the data.

Returns None

post_sequences (*sequences: List[cognite.client.experimental.sequences.Sequence]*) → `cognite.client.experimental.sequences.Sequence`

Create a new time series.

Parameters **sequences** (*list[test_experimental.dto.Sequence]*) – List of sequence data transfer objects to create.

Returns The created sequence

Return type `client.test_experimental.sequences.Sequence`

3.4.2 Data Objects

```
class cognite.client.experimental.sequences.Column (id: int = None, name: str = None, external_id: str = None, value_type: str = None, metadata: dict = None)
```

Bases: `object`

Data transfer object for a column.

Parameters

- **id** (*int*) – ID of the column.
- **name** (*str*) – Name of the column.
- **external_id** (*str*) – External ID of the column.
- **value_type** (*str*) – Data type of the column.
- **metadata** (*dict*) – Custom, application specific metadata. String key -> String Value.

static from_JSON (*the_column: dict*)

class `cognite.client.experimental.sequences.Row` (*row_number: int, values: List[cognite.client.experimental.sequences.RowValue]*)

Bases: `object`

Data transfer object for a row of data in a sequence.

Parameters

- **row_number** (*int*) – The row number for this row.
- **values** (*list*) – The values in this row.

static from_JSON (*the_row: dict*)

get_row_as_csv ()

class `cognite.client.experimental.sequences.RowValue` (*column_id: int, value: str*)

Bases: `object`

Data transfer object for the value in a row in a sequence.

Parameters

- **column_id** (*int*) – The ID of the column that this value is for.
- **value** (*str*) – The actual value.

static from_JSON (*the_row_value: dict*)

class `cognite.client.experimental.sequences.Sequence` (*id: int = None, name: str = None, external_id: str = None, asset_id: int = None, columns: List[cognite.client.experimental.sequences.Column] = None, description: str = None, metadata: dict = None*)

Bases: `object`

Data transfer object for a sequence.

Parameters

- **id** (*int*) – ID of the sequence.
- **name** (*str*) – Name of the sequence.
- **external_id** (*str*) – External ID of the sequence.
- **asset_id** (*int*) – ID of the asset the sequence is connected to, if any.
- **columns** (*List[Column]*) – List of columns in the sequence.
- **description** (*str*) – Description of the sequence.
- **metadata** (*dict*) – Custom, application specific metadata. String key -> String Value.

static from_JSON (*the_sequence: dict*)

```
class cognite.client.experimental.sequences.SequenceDataRequest (inclusive_from:
                                                                    int,   inclu-
                                                                    sive_to:  int,
                                                                    limit:   int
                                                                    = 100,   col-
                                                                    umn_ids:
                                                                    List[int]  =
                                                                    None)
```

Bases: object

Data transfer object for requesting sequence data.

Parameters

- **inclusive_from** (*int*) – Row number to get from (inclusive).
- **inclusive_to** (*int*) – Row number to get to (inclusive).
- **limit** (*int*) – How many rows to return.
- **column_ids** (*List[int]*) – ids of the columns to get data for.

```
class cognite.client.experimental.sequences.SequenceDataResponse (rows:
                                                                    List[cognite.client.experimental.seque
```

Bases: object

Data transfer object for the data in a sequence, used when receiving data.

Parameters **rows** (*list*) – List of rows with the data.

static from_JSON (*the_data: dict*)

to_json ()

Returns data as a json object

to_pandas ()

Returns data as a pandas dataframe

Data Transfer Service

```
class cognite.data_transfer_service.DataSpec (time_series_data_specs:
                                         List[cognite.data_transfer_service.TimeSeriesDataSpec]
                                         = None, files_data_spec: cog-
                                         nite.data_transfer_service.FilesDataSpec
                                         = None)
```

Bases: object

Object for specifying data when querying CDP.

Parameters

- **time_series_data_specs** (*List[TimeSeriesDataSpec]*) – Time Series data specs
- **files_data_spec** (*FilesDataSpec*) – Files data spec

Raises *DataSpecValidationError* – An error occurred while validating the data spec

```
classmethod from_JSON (json_repr)
```

```
to_JSON ()
```

```
exception cognite.data_transfer_service.DataSpecValidationError
```

Bases: Exception

```
class cognite.data_transfer_service.DataTransferService (data_spec: cog-
                                                                nite.data_transfer_service.DataSpec,
                                                                api_key: str = None,
                                                                cookies: Dict = None,
                                                                num_of_workers: int =
                                                                10)
```

Bases: object

Create a Data Transfer Service object.

Fetch timeseries from the api.

```
get_dataframe (label: str = 'default', drop_agg_suffix: bool = True)
```

get_dataframes (*drop_agg_suffix: bool = True*)

Return a dictionary of dataframes indexed by label - one per data spec.

Parameters **drop_agg_suffix** (*bool*) – If a time series has only one aggregate, drop the `<agg-func>` suffix on those column names.

Returns A label-indexed dictionary of data frames.

Return type Dict[str, pd.DataFrame]

get_file (*name*)

Return files by name as specified in the DataSpec

Parameters **name** (*str*) – Name of file

get_time_series_name (*ts_label: str, dataframe_label: str = 'default'*)

class `cognite.data_transfer_service.FilesDataSpec` (*file_ids: Dict[str, int]*)

Bases: object

Object for specifying data from the Files API when using a data spec.

Parameters **file_ids** (*Dict[str, int]*) – Dictionary of fileNames -> fileIds

class `cognite.data_transfer_service.TimeSeries` (*id: int, aggregates: List[str] = None, missing_data_strategy: str = None, label: str = None*)

Bases: object

Object for specifying a specific time series from the TimeSeries API when using a data spec.

Parameters

- **id** (*int*) – id of time series to retrieve
- **aggregates** (*List[str]*) – Local aggregate functions to apply
- **missing_data_strategy** (*str*) – Missing data strategy to apply
- **label** (*str*) – name of the column in the resulting data frame when passed to data transfer service.

Examples

When you supply a label the resulting data frames produced by data transfer service will use the label as column names.

```
class cognite.data_transfer_service.TimeSeriesDataSpec (time_series: List[cognite.data_transfer_service.TimeSeries], aggregates: List[str], granularity: str, missing_data_strategy: str = None, start: Union[str, int, datetime.datetime] = None, end: Union[str, int, datetime.datetime] = None, label: str = None)
```

Bases: object

Object for specifying data from the TimeSeries API when using a data spec.

Parameters

- **time_series** (*List[data_transfer_service.TimeSeries]*) – Time series
- **aggregates** (*List[str]*) – List of aggregate functions
- **granularity** (*str*) – Granularity of aggregates
- **missing_data_strategy** (*str*) – Missing data strategy to apply, can be “linearInterpolation” or “ffill”
- **start** (*Union[str, int, datetime]*) – Start time
- **end** (*Union[str, int, datetime]*) – end time
- **label** (*str*) – Label for this data spec

Examples

When you specify a label, data transfer service will exhibit the following behaviour:

```
ts_data_spec = TimeSeriesDataSpec(..., label="some_label")
data_spec = DataSpec([ts_data_spec], ...)
dts = DataTransferService(data_spec)
dataframes = dts.get_dataframes
my_df = dataframes["some_label"]
```


C

`cognite.client.exceptions`, 5
`cognite.client.experimental.analytics.models`,
37
`cognite.client.experimental.datapoints`,
40
`cognite.client.experimental.sequences`,
41
`cognite.client.experimental.time_series`,
39
`cognite.client.stable.assets`, 10
`cognite.client.stable.datapoints`, 16
`cognite.client.stable.events`, 20
`cognite.client.stable.files`, 24
`cognite.client.stable.login`, 25
`cognite.client.stable.raw`, 28
`cognite.client.stable.tagmatching`, 29
`cognite.client.stable.time_series`, 31
`cognite.data_transfer_service`, 45

A

APIError, 5
 Asset (class in `cognite.client.stable.assets`), 10
 AssetListResponse (class in `cognite.client.stable.assets`), 10
 AssetResponse (class in `cognite.client.stable.assets`), 11
 AssetsClient (class in `cognite.client.stable.assets`), 7

C

`cognite.client.exceptions` (module), 5
`cognite.client.experimental.analytics.models` (module), 37
`cognite.client.experimental.datapoints` (module), 40
`cognite.client.experimental.sequences` (module), 41
`cognite.client.experimental.time_series` (module), 39
`cognite.client.stable.assets` (module), 10
`cognite.client.stable.datapoints` (module), 16
`cognite.client.stable.events` (module), 20
`cognite.client.stable.files` (module), 24
`cognite.client.stable.login` (module), 25
`cognite.client.stable.raw` (module), 28
`cognite.client.stable.tagmatching` (module), 29
`cognite.client.stable.time_series` (module), 31
`cognite.data_transfer_service` (module), 45
 CogniteClient (class in `cognite`), 3
 CogniteCollectionResponse (class in `cognite.client._api_client`), 5
 CogniteResponse (class in `cognite.client._api_client`), 4
 Column (class in `cognite.client.experimental.sequences`), 41
 create_databases() (`cognite.client.stable.raw.RawClient` method), 26
 create_model() (`cognite.client.experimental.analytics.models.ModelsClient` method), 33
 create_rows() (`cognite.client.stable.raw.RawClient` method), 26
 create_schedule() (`cognite.client.experimental.analytics.models.ModelsClient` method), 33

create_source_package() (`cognite.client.experimental.analytics.models.ModelsClient` method), 34
 create_tables() (`cognite.client.stable.raw.RawClient` method), 26

D

Datapoint (class in `cognite.client.stable.datapoints`), 16
 DatapointsClient (class in `cognite.client.experimental.datapoints`), 39
 DatapointsClient (class in `cognite.client.stable.datapoints`), 11
 DatapointsQuery (class in `cognite.client.stable.datapoints`), 16
 DatapointsResponse (class in `cognite.client.experimental.datapoints`), 40
 DatapointsResponse (class in `cognite.client.stable.datapoints`), 16
 DatapointsResponseIterator (class in `cognite.client.stable.datapoints`), 17
 DataSpec (class in `cognite.data_transfer_service`), 45
 DataSpecValidationError, 45
 DataTransferService (class in `cognite.data_transfer_service`), 45
 delete() (`cognite.CogniteClient` method), 4
 delete_assets() (`cognite.client.stable.assets.AssetsClient` method), 7
 delete_databases() (`cognite.client.stable.raw.RawClient` method), 26
 delete_events() (`cognite.client.stable.events.EventsClient` method), 17
 delete_files() (`cognite.client.stable.files.FilesClient` method), 21
 delete_model() (`cognite.client.experimental.analytics.models.ModelsClient` method), 35
 delete_rows() (`cognite.client.stable.raw.RawClient` method), 26
 delete_schedule() (`cognite.client.experimental.analytics.models.ModelsClient` method), 35

delete_sequence_by_id() (cognite.client.experimental.sequences.SequencesClient method), 40

delete_source_package() (cognite.client.experimental.analytics.models.ModelsClient method), 35

delete_tables() (cognite.client.stable.raw.RawClient method), 27

delete_time_series() (cognite.client.stable.time_series.TimeSeriesClient method), 29

delete_time_series_by_id() (cognite.client.experimental.time_series.TimeSeriesClient method), 37

delete_version() (cognite.client.experimental.analytics.models.ModelsClient method), 35

download_file() (cognite.client.stable.files.FilesClient method), 21

E

Event (class in cognite.client.stable.events), 20

EventListResponse (class in cognite.client.stable.events), 20

EventResponse (class in cognite.client.stable.events), 21

EventsClient (class in cognite.client.stable.events), 17

F

FileInfoResponse (class in cognite.client.stable.files), 24

FileListResponse (class in cognite.client.stable.files), 24

FilesClient (class in cognite.client.stable.files), 21

FilesDataSpec (class in cognite.data_transfer_service), 46

from_JSON() (cognite.client.experimental.sequences.Column static method), 42

from_JSON() (cognite.client.experimental.sequences.Row static method), 42

from_JSON() (cognite.client.experimental.sequences.RowValue static method), 42

from_JSON() (cognite.client.experimental.sequences.Sequence static method), 42

from_JSON() (cognite.client.experimental.sequences.SequenceDataResponse static method), 43

from_JSON() (cognite.data_transfer_service.DataSpec class method), 45

G

get() (cognite.CogniteClient method), 4

get_asset() (cognite.client.stable.assets.AssetsClient method), 7

get_asset_subtree() (cognite.client.stable.assets.AssetsClient method), 8

get_assets() (cognite.client.stable.assets.AssetsClient method), 8

get_data_from_sequence() (cognite.client.experimental.sequences.SequencesClient method), 40

get_databases() (cognite.client.stable.raw.RawClient method), 27

get_dataframe() (cognite.data_transfer_service.DataTransferService method), 45

get_dataframes() (cognite.data_transfer_service.DataTransferService method), 45

get_datapoints() (cognite.client.experimental.datapoints.DatapointsClient method), 39

get_datapoints() (cognite.client.stable.datapoints.DatapointsClient method), 11

get_datapoints_frame() (cognite.client.stable.datapoints.DatapointsClient method), 12

get_event() (cognite.client.stable.events.EventsClient method), 17

get_events() (cognite.client.stable.events.EventsClient method), 18

get_file() (cognite.data_transfer_service.DataTransferService method), 46

get_file_info() (cognite.client.stable.files.FilesClient method), 22

get_latest() (cognite.client.stable.datapoints.DatapointsClient method), 13

get_model() (cognite.client.experimental.analytics.models.ModelsClient method), 35

get_models() (cognite.client.experimental.analytics.models.ModelsClient method), 35

get_multi_time_series_datapoints() (cognite.client.stable.datapoints.DatapointsClient method), 13

get_multiple_time_series_by_id() (cognite.client.experimental.time_series.TimeSeriesClient method), 37

get_row() (cognite.client.stable.raw.RawClient method), 27

get_row_as_csv() (cognite.client.experimental.sequences.Row method), 42

get_rows() (cognite.client.stable.raw.RawClient method), 27

get_schedule() (cognite.client.experimental.analytics.models.ModelsClient method), 35

get_schedules() (cognite.client.experimental.analytics.models.ModelsClient method), 35

get_sequence_by_external_id() (cognite.client.experimental.sequences.SequencesClient method), 41

get_sequence_by_id() (cognite.client.experimental.sequences.SequencesClient method), 41

get_source_package() (cognite.client.stable.raw.RawClient method), 27

nite.client.experimental.analytics.models.ModelsClient
 method), 36
 next_cursor() (cognite.client.stable.datapoints.LatestDatapointResponse
 method), 17
 get_source_packages() (cog- next_cursor() (cognite.client.stable.events.EventListResponse
 nite.client.experimental.analytics.models.ModelsClient method), 20
 method), 36 next_cursor() (cognite.client.stable.events.EventResponse
 method), 21
 get_tables() (cognite.client.stable.raw.RawClient next_cursor() (cognite.client.stable.files.FileInfoResponse
 method), 27 method), 24
 next_cursor() (cognite.client.stable.files.FileListResponse
 method), 24
 get_time_series() (cog- next_cursor() (cognite.client.stable.files.FileListResponse
 nite.client.stable.time_series.TimeSeriesClient method), 24
 method), 30 next_cursor() (cognite.client.stable.login.LoginStatusResponse
 method), 25
 get_time_series_by_id() (cog- next_cursor() (cognite.client.stable.raw.RawResponse
 nite.client.experimental.time_series.TimeSeriesClient method), 28
 method), 37 next_cursor() (cognite.client.stable.tagmatching.TagMatchingResponse
 method), 29
 get_time_series_name() (cog- next_cursor() (cognite.client.stable.tagmatching.TagMatchingResponse
 nite.data_transfer_service.DataTransferService method), 29
 method), 46
 get_version() (cognite.client.experimental.analytics.models.ModelsClient) (cognite.client.stable.time_series.TimeSeriesListResponse
 method), 36 method), 32
 get_versions() (cognite.client.experimental.analytics.models.ModelsClient) (cognite.client.stable.time_series.TimeSeriesResponse
 method), 36 method), 32

L

LatestDatapointResponse (class in cog-
 nite.client.stable.datapoints), 17
 list_files() (cognite.client.stable.files.FilesClient method),
 22
 live_data_generator() (cog-
 nite.client.stable.datapoints.DatapointsClient
 method), 14
 logged_in (cognite.client.stable.login.LoginStatusResponse
 attribute), 25
 LoginClient (class in cognite.client.stable.login), 25
 LoginStatusResponse (class in cog-
 nite.client.stable.login), 25

M

ModelsClient (class in cog-
 nite.client.experimental.analytics.models),
 33

N

next_cursor() (cognite.client._api_client.CogniteResponse
 method), 5
 next_cursor() (cognite.client.experimental.datapoints.DatapointResponse
 method), 40
 next_cursor() (cognite.client.experimental.time_series.TimeSeriesResponse
 method), 39
 next_cursor() (cognite.client.stable.assets.AssetListResponse
 method), 11
 next_cursor() (cognite.client.stable.assets.AssetResponse
 method), 11
 next_cursor() (cognite.client.stable.datapoints.DatapointsResponse
 method), 16
 next_cursor() (cognite.client.stable.datapoints.DatapointsResponse
 method), 16
 next_cursor() (cognite.client.stable.events.EventListResponse
 method), 20
 next_cursor() (cognite.client.stable.events.EventResponse
 method), 21
 next_cursor() (cognite.client.stable.files.FileInfoResponse
 method), 24
 next_cursor() (cognite.client.stable.files.FileListResponse
 method), 24
 next_cursor() (cognite.client.stable.login.LoginStatusResponse
 method), 25
 next_cursor() (cognite.client.stable.raw.RawResponse
 method), 28
 next_cursor() (cognite.client.stable.tagmatching.TagMatchingResponse
 method), 29
 ModelsClient (cognite.client.stable.time_series.TimeSeriesListResponse
 method), 32
 ModelsClient (cognite.client.stable.time_series.TimeSeriesResponse
 method), 32

O

online_predict() (cognite.client.experimental.analytics.models.ModelsClient
 method), 36

P

post() (cognite.CogniteClient method), 4
 post_assets() (cognite.client.stable.assets.AssetsClient
 method), 9
 post_data_to_sequence() (cog-
 nite.client.experimental.sequences.SequencesClient
 method), 41
 post_datapoints() (cog-
 nite.client.stable.datapoints.DatapointsClient
 method), 14
 post_datapoints_frame() (cog-
 nite.client.stable.datapoints.DatapointsClient
 method), 14
 post_events() (cognite.client.stable.events.EventsClient
 method), 18
 post_multi_time_series_datapoints() (cog-
 nite.client.stable.datapoints.DatapointsClient
 method), 15
 post_sequences() (cognite.client.experimental.sequences.SequencesClient
 method), 41
 post_time_series() (cog-
 nite.client.stable.time_series.TimeSeriesClient
 method), 30
 previous_cursor() (cog-
 nite.client._api_client.CogniteResponse
 method), 5
 previous_cursor() (cog-
 nite.client.experimental.datapoints.DatapointsResponse
 method), 40

previous_cursor() (cognite.client.experimental.time_series.TimeSeriesResponseValue method), 39

previous_cursor() (cognite.client.stable.assets.AssetListResponse method), 11

previous_cursor() (cognite.client.stable.assets.AssetResponse method), 11

previous_cursor() (cognite.client.stable.datapoints.DatapointsResponse method), 16

previous_cursor() (cognite.client.stable.datapoints.LatestDatapointResponse method), 17

previous_cursor() (cognite.client.stable.events.EventListResponse method), 20

previous_cursor() (cognite.client.stable.events.EventResponse method), 21

previous_cursor() (cognite.client.stable.files.FileInfoResponse method), 24

previous_cursor() (cognite.client.stable.files.FileListResponse method), 24

previous_cursor() (cognite.client.stable.login.LoginStatusResponse method), 25

previous_cursor() (cognite.client.stable.raw.RawResponse method), 28

previous_cursor() (cognite.client.stable.tagmatching.TagMatchingResponse method), 29

previous_cursor() (cognite.client.stable.time_series.TimeSeriesListResponse method), 32

previous_cursor() (cognite.client.stable.time_series.TimeSeriesResponse method), 32

project (cognite.client.stable.login.LoginStatusResponse attribute), 25

project_id (cognite.client.stable.login.LoginStatusResponse attribute), 25

put() (cognite.CogniteClient method), 4

R

RawClient (class in cognite.client.stable.raw), 26

RawResponse (class in cognite.client.stable.raw), 28

RawRow (class in cognite.client.stable.raw), 28

repr_json() (cognite.client.stable.raw.RawRow method), 28

Row (class in cognite.client.experimental.sequences), 42

RowValue (class in cognite.client.experimental.sequences), 42

S

search_for_assets() (cognite.client.stable.assets.AssetsClient method), 9

search_for_events() (cognite.client.stable.events.EventsClient method), 19

search_for_time_series() (cognite.client.experimental.time_series.TimeSeriesClient method), 38

Sequence (class in cognite.client.experimental.sequences), 42

SequenceDataRequest (class in cognite.client.experimental.sequences), 42

SequenceDataResponse (class in cognite.client.experimental.sequences), 43

SequencesClient (class in cognite.client.experimental.sequences), 40

status() (cognite.client.stable.login.LoginClient method), 25

T

tag_matching() (cognite.client.stable.tagmatching.TagMatchingClient method), 28

TagMatchingClient (class in cognite.client.stable.tagmatching), 28

TagMatchingResponse (class in cognite.client.stable.tagmatching), 29

TimeSeries (class in cognite.client.stable.time_series), 31

TimeSeries (class in cognite.data_transfer_service), 46

TimeSeriesClient (class in cognite.client.experimental.time_series), 37

TimeSeriesClient (class in cognite.client.stable.time_series), 29

TimeSeriesDataSpec (class in cognite.data_transfer_service), 46

TimeSeriesListResponse (class in cognite.client.stable.time_series), 32

TimeSeriesResponse (class in cognite.client.experimental.time_series), 39

TimeseriesWithDatapoints (class in cognite.client.stable.datapoints), 17

to_json() (cognite.client._api_client.CogniteCollectionResponse method), 5

to_json() (cognite.client._api_client.CogniteResponse method), 5

to_json() (cognite.client.experimental.datapoints.DatapointsResponse method), 40

- to_json() (cognite.client.experimental.sequences.SequenceDataResponse method), 43
- to_json() (cognite.client.experimental.time_series.TimeSeriesResponse method), 39
- to_json() (cognite.client.stable.assets.AssetListResponse method), 11
- to_json() (cognite.client.stable.assets.AssetResponse method), 11
- to_json() (cognite.client.stable.datapoints.DatapointsResponse method), 16
- to_json() (cognite.client.stable.datapoints.LatestDatapointResponse method), 17
- to_json() (cognite.client.stable.events.EventListResponse method), 20
- to_json() (cognite.client.stable.events.EventResponse method), 21
- to_json() (cognite.client.stable.files.FileInfoResponse method), 24
- to_json() (cognite.client.stable.files.FileListResponse method), 24
- to_json() (cognite.client.stable.login.LoginStatusResponse method), 25
- to_json() (cognite.client.stable.raw.RawResponse method), 28
- to_json() (cognite.client.stable.tagmatching.TagMatchingResponse method), 29
- to_json() (cognite.client.stable.time_series.TimeSeriesListResponse method), 32
- to_json() (cognite.client.stable.time_series.TimeSeriesResponse method), 32
- to_JSON() (cognite.data_transfer_service.DataSpec method), 45
- to_list() (cognite.client.stable.tagmatching.TagMatchingResponse method), 29
- to_pandas() (cognite.client.experimental.datapoints.DatapointsResponse method), 40
- to_pandas() (cognite.client.experimental.sequences.SequenceDataResponse method), 43
- to_pandas() (cognite.client.experimental.time_series.TimeSeriesResponse method), 39
- to_pandas() (cognite.client.stable.assets.AssetListResponse method), 11
- to_pandas() (cognite.client.stable.assets.AssetResponse method), 11
- to_pandas() (cognite.client.stable.datapoints.DatapointsResponse method), 16
- to_pandas() (cognite.client.stable.datapoints.LatestDatapointResponse method), 17
- to_pandas() (cognite.client.stable.events.EventListResponse method), 21
- to_pandas() (cognite.client.stable.events.EventResponse method), 21
- to_pandas() (cognite.client.stable.files.FileInfoResponse method), 24
- to_pandas() (cognite.client.stable.files.FileListResponse method), 24
- to_pandas() (cognite.client.stable.tagmatching.TagMatchingResponse method), 29
- to_pandas() (cognite.client.stable.time_series.TimeSeriesListResponse method), 32
- to_pandas() (cognite.client.stable.time_series.TimeSeriesResponse method), 32
- update_time_series() (cognite.client.stable.time_series.TimeSeriesClient method), 31
- upload_file() (cognite.client.stable.files.FilesClient method), 23
- user (cognite.client.stable.login.LoginStatusResponse attribute), 25
- model_version() (cognite.client.experimental.analytics.models.ModelsClient method), 36

U