questradeapi Documentation

Release

Antoine Viscardi

Nov 07, 2018

Contents:

CHAPTER 1

Installation

To install QuestradeAPI, you can use pipenv (or pip, of course):

```
pipenv install questradeapi
```

CHAPTER 2

Quick Start

2.1 Prerequisites

If you are reading this quick start guide, it is assumed that you have an active Questrade trading account with which to use the API. If this is not the case, then I don't know what you are doing here ¯_()_/¯.

2.2 Setup

In order to acces Questrade's API, you need to authorize your application to use the API through your account. This can be done by following this guide from Questrade. Once this is done, you should be able to generate refresh tokens. This is important because you will need to feed one of those token to the Session in order to be able to perform API calls.

2.3 Using the API

A `Sessions` is used to make the different API calls. Given an initial refresh token, it automatically deals with the redeeming process to get access tokens.

The following code snippet initializes a session with a refresh token and performs some simple API calls.

```
import questradeapi as qapi

refresh_token = input('Refresh Token: ')
sess = qapi.Session(refresh_token)

id = sess.get_accounts()['accounts'][0]['number'] #Fetch first account ID.
positions = sess.get_positions(id)
balances = sess.get_balances(id)
```

CHAPTER 3

API

Thorough documentation of the different available API calls can be found on Questrade's website here.

3.1 Enumeration Values

Some API calls' parameters only accept specific string values. A list of those enumerations and their description can be found here on Questrade's website.

3.2 Data Structures

Some API calls' parameters requires `dict` of a specific structure to be passed. These data structures and utility methods to create them are documented in the *Data Structures* section.

3.3 Account Calls

`Session.`**`get_time`**`()`
Retrieve current server time.

Returns Dictionary containing the response properties.

Return type `dict`

`Session.`**`get_accounts`**`()`
Retrieves the accounts associated with the user on behalf of which the API client is authorized.

Returns Dictionary containing the response properties.

Return type `dict`

`Session.`**`get_positions`**`(`*id*`)`
Retrives positions in a specified account.

Parameters **id** (str) – Account number

Returns Dictionary containing the response properties.

Return type dict

Session.**get_balances**(*id*)
Retrieves per-currency and combined balances for a specified account.

Parameters **id** (str) – Account number

Returns Dictionary containing the response properties.

Return type dict

Session.**get_executions**(*id*, *start_time=None*, *end_time=None*)
Retrieves executions for a specific account.

Parameters

- **id** (str) – Account number
- **start_time** (datetime, optional) – Start of the time range. Defaults to today 00:00am.
- **end_time** (datetime, optional) – End of the time range. Defaults to todat 11:59pm.

Returns Dictionary containing the response properties.

Return type dict

Session.**get_orders**(*id*, *state_filter=None*, *start_time=None*, *end_time=None*, *order_ids=None*)
Retrieves orders for a specified account.

Parameters

- **id** (str) – Account number
- **state_filter** (str, {'All', 'Open', 'Close'}) – Retreive all, active or closed orders.
- **start_time** (datetime, optional) – Start of the time range. Defaults to today 00:00am.
- **end_time** (datetime, optional) – End of the time range. Defaults to todat 11:59pm.
- **order_ids** (*int, optional*) – Retrieve specific orders details.

Note: More details on allowed *state_filter* values can be found here.

Returns Dictionary containing the response properties.

Return type dict

Session.**get_activities**(*id*, *start_time=None*, *end_time=None*)
Retrieve account activities, including cash transactons, dividends, trades, etc.

Parameters

- **id** (list of str) – Account number.
- **start_time** (datetime, optional) – Start of the time range. Defaults to today 12:00am.
- **end_time** (endtime, optional) – End of the time range Defaults to today 11:59pm.

Returns Dictionary containing the response properties.

Return type dict

3.4 Market Calls

Session.**get_symbols**(*names=None*, *ids=None*, *id=None*)
Retrieves detailed information about one or more symbol.

Parameters

- **name** (list of str) – List of symbol names.
- **ids** (list of int) – List of symbol ids.
- **id** (int) – Internal symbol identifier. Mutually exclusive with 'ids' parameter.

Returns

- dict – Dictionary containing the response properties.
- *Either list of names or ids can be specified, but not both. If 'names'*
- *is specified, it takes precedence over 'ids', which takes precedence*
- *over 'id'.*

Session.**get_symbols_search**(*prefix*, *offset=None*)
Retrieves symbol(s) using several search criteria.

Parameters

- **prefix** (str) – Prefix of a symbol or any word in the description.
- **offset** (int) – Offset in number of records from the beginning of a result set.

Returns Dictionary containing the response properties.

Return type dict

Session.**get_option_chain**(*id*)
Retrieves an option chain for a particular underlying symbol.

Parameters **id** (int) – Internal symbol identifier.

Session.**get_markets**()
Retrieves information about supported markets.

Returns Dictionary containing the response properties.

Return type dict

Session.**get_quotes**(*id=None*, *ids=None*)
Retrieves a single Level 1 market data quote for one or more symbols.

Parameters

- **id** (int) – Internal symbol identifier (mutually exclusive with 'ids' argument).
- **ids** (list of int) – List of symbol ids.

Returns Dictionary containing the response properties.

Return type dict

Session.**get_quotes_options**(*filters=None*, *ids=None*)
Retrieves a single Level 1 market data quote and Greek data for one or more option symbols.

Parameters

- **filters** (list of dict) – List of OptionIdFilter structures. See the utility function questradeapi.utils.create_option_id_filter().

- **ids** (list of int) – List of option IDs.

Returns Dictionary containing the response properties.

Return type dict

Session.**get_quotes_strategies**(*variants*)
Retrieve a calculated L1 market data quote for a single or many multi-leg strategies.

Parameters **variants** (list of dict) – List of StrategyVariantsRequest structures. See utility function questradeapi.utils.create_strategy_variant_request().

Returns Dictionary containing the response properties.

Return type dict

Session.**get_candles**(*id*, *start_time*, *end_time*, *interval*)
Retrieves historical market data in the form of OHLC candlesticks for a specified symbol. This call is limited to returning 2,000 candlesticks in a single response.

Parameters

- **id** (int) – Internal symbol indentifier.
- **startTime** (datetime) – Beginning of the candlestick range.
- **endTime** (datetime) – End of the candlestick range.
- **interval** (str, {'OneMinute', 'TwoMinutes', 'ThreeMinutes', 'FourMinutes', 'FiveMinutes', 'TenMinutes', 'FifteenMinutes', 'TwentyMinutes', 'HalfHour', 'OneHour', 'TwoHours', 'FourHours', 'OneDay', 'OneWeek', 'OneMonth', 'OneYear'}) – Interval of a single candlestick.

Note: More details on allowed *interval* values can be found here

---

Returns Dictionary containing the response properties.

Return type dict

3.5 Order Calls

Session.**post_order**(*account_id*, *symbol_id*, *quantity*, *iceberg_quantity*, *limit_price*, *stop_price*, *all_or_none*, *anonymous*, *order_type*, *time_in_force*, *action*, *primary_route*, *secondary_route*, *order_id=None*, *impact=False*)
Allows to place/replace or estimate the impact of an order against a certain account.

Parameters

- **id** (str) – Account number against which the order is submited.
- **order_id** (int) – Order id of the order to be replaced.
- **symbol_id** (int) – Internal symbol identifier.
- **quantity** (int) – Order quantity.
- **iceberg_quantity** (int) – Iceberg instruction quantity.
- **limit_price** (double) – Limit price.
- **stop_price** (double) – Stop price.

- **all_or_none** (bool) – Whether all are none is enabled.
- **anonymous** (bool) – Wheter anonymous is enabled.
- **order_type** (str, 'Market', 'Limit', 'Stop', 'StopLimit', 'TrailStopInPercentage', 'TrailStopInDollar', 'TrailStopLimitInPercentage', 'TrailStopLimitInDollar', 'LimitOnOpen', 'LimitOnClose'}) – Order type.
- **time_in_force** (str, {'Day', 'GoodTillCanceled', 'GoodTillExtendedDay', 'GoodTillDate', 'ImmediateOrCancel', FillOrKill}) – Order duration.
- **action** (str, {'Buy', 'Sell'}) – Order side.
- **primary_route** (str, optional) – Primary order route. Defaults to 'AUTO'.
- **secondary_route** (str, optional) – Secondary order route. Defaults to 'AUTO'.
- **impact** (bool) – Calculate impact instead of placing order.

Returns Dictionary containing the response properties.

Return type dict

Session.**delete_order**(*order_id*)

Allows to cancel an existing order.

Parameters

- **account_id** (str) – Account number.
- **order_id** (int) – Internal identifier of the order.

Returns Dictionary containing the response properties.

Return type dict

Session.**post_bracket_order**(*account_id*, *symbol_id*, *primary_route*, *secondary_route*, *components*, *impact=False*)

Allows to place/replace or estimate the impact of a bracket order against a certain account.

Parameters

- **account_id** (str) – Account number.
- **symbol_id** (int) – Internal symbol identifier.
- **primary_route** (str, optional) – Primary order route. Defaults to 'AUTO'.
- **secondary_route** (str, optional) – Secondary order route. Defaults to 'AUTO'.
- **components** (list of dict) – List of Bracket Order Components. See the utility function questradeapi.utils.create_bracket_order_component().
- **impact** (bool) – Calculate impact instead of placing order.

Returns Dictionary containing the response properties.

Return type dict

Session.**post_multi_leg_strategy_order**(*account_id*, *symbol_id*, *limit_price*, *order_type*, *time_in_force*, *primary_route*, *secondary_route*, *legs*, *strategy*, *impact=False*)

Allows to place/replace or estimate the impact of a multi-leg strategy against a certain account.

Parameters

- **account_id** (str) – Account number.
- **symbol_id** (int) – Internal symbol identifier.

- **limit_price** (double) – Limit price.
- **order_type** (str, 'Market', 'Limit', 'Stop', 'StopLimit', 'TrailStopInPercentage', 'TrailStopInDollar', 'TrailStopLimitInPercentage', 'TrailStopLimitInDollar', 'LimitOnOpen', 'LimitOnClose'}) – Order type.
- **time_in_force** (str, {'Day', 'GoodTillCanceled', 'GoodTillExtendedDay', 'GoodTillDate', 'ImmediateOrCancel', FillOrKill}) – Order duration.
- **primary_route** (str, optional) – Primary order route. Defaults to 'AUTO'.
- **secondary_route** (str, optional) – Secondary order route. Defaults to 'AUTO'.
- **legs** (list of dict) – List of InsertOrderLegData structures. See the utility function questradeapi.utils.create_insert_order_leg_data().
- **strategy** (str, {'CoveredCall', 'MarriedPuts', 'VerticalCallSpread', 'VerticalPutSpread', 'CalendarCallSpread', 'CalendarPutSpread', 'DiagonalCallSpread', 'DiagonalPutSpread', 'Collar', 'Straddle', 'Strangle', 'ButterflyCall', 'ButterflyPut', 'IronButterfly', 'CondorCall', 'Custom'}) – Strategy type.
- **impact** (bool) – Calculate impact instead of placing order.

Note: More details on allowed *order_type* values can be found here

Note: More details on allowed *time_in_force* values can be found here

Note: More details on allowed *strategy* values can be found here

Returns Dictionary containing the response properties.

Return type dict

CHAPTER 4

Data Structures

CHAPTER 5

TODOs

CHAPTER 6

Indices and tables

- genindex
- modindex
- search

Python Module Index

q

Index