tdameritrade-cli

Release 0.1.1

Joe Castagneri

Apr 26, 2019

CONTENTS:

CHAPTER

ONE

INSTALLATION INSTRUCTIONS

1.1 System Dependencies

Python 3.6 or 3.7.

1.2 Linux

Create a virtual environment and install with:

```
pip install tdameritrade-cli
```

If your virtual environment is located at `$HOME/.virtualenvs/NAME_OF_VENV`, then add the CLI to your path with:

```
ln -s $HOME/.virtualenvs/NAME_OF_VENV/bin/tda-cli $HOME/.local/bin/tda-cli
```

And open a new shell. You can test the installation with `tda-cli --help`

To update the package, navigate to the virtual environment directory and run:

```
bin/pip install --upgrade tdameritrade-cli
```

1.3 Windows

1. If not already present on your system, install Python 3 here.
2. Open a `CMD` shell, create a new directory where you'd like to install the CLI, enter it, and create a virtual environment:

   ```
   mkdir td-cli
   cd td-cli
   py -m venv .
   ```

3. Install the CLI into the virtual environment:

   ```
   Scripts\pip install tdameritrade-cli
   ```

4. Add the package's Scripts directory (located at `tda-cli\Scripts`) to your path (instructions).
5. Open a new shell and test the installation with `tda-cli --help`.

6. To update the package, navigate to the virtual environment directory and run:

```
Scripts\pip install --update tdameritrade-cli
```

1.4 Usage

For usage instructions, consult the quickstart.

CHAPTER

TWO

QUICKSTART

2.1 Before You Begin:

This CLI is build on the TDAmeritrade Client PyPi package.

The package assumes you or someone who trusts you has set up a TDAmeritrade app through their developer portal. If you haven't done this, see TDAmeritrade's getting started guide.

To authenticate against the TDAmeritrade API, you need:

1. An OAuth redirect URI
2. An OAuth user ID
3. Your TDAmeritrade account number

You create these first two items when you register a new app with TDAmeritrade.

CHAPTER

THREE

WAYS TO AUTHENTICATE

You can authenticate by either passing in the three items above individually (with the `-a`, `-o`, and `-r` options), or by writing a json object containing the same information:

```
{
    "acct_number": 1234567890,
    "oauth_user_id": MYOAUTHID,
    "redirect_uri": http://127.0.0.1:8080
}
```

Then, you can point the CLI to this location of this file with the `-j` option.

Execute `tda-cli --help` for full help text.

3.1 Current Uses:

The CLI has the following uses:

1. Print your positions data to the console:

   ```
   tda-cli -j PATH_TO_JSON list-positions
   ```

2. Write your positions data to an ods file:

   ```
   tda-cli -j PATH_TO_JSON write-positions PATH_TO_ODS_OUTPUT
   ```

When writing to an ods, the CLI will generate a cover sheet with current positions, and update current liquidation throughout the calendar year. Subsequent sheets are time data of each position which is appended every time the command is run against the same sheet. This functionality is meant to be run automatically and frequently to generate a local aggregate of your investment data.

CHAPTER

FOUR

DOCSTRINGS

4.1 tdameritrade_cli.tools package

4.1.1 Submodules

4.1.2 tdameritrade_cli.tools.client_factory module

class tdameritrade_cli.tools.client_factory.**ClientFactory**(*json_config: str = None*, *acct_number: int = None*, *oauth_user_id: str = None*, *redirect_uri: str = None*)

Bases: object

from_config()

Create a TDClient from passed parameters

Returns An authenticated TDClient.

from_json()

Create a TDClient from a json_config

Returns An authenticated TDClient.

Raises

- **IsADirectoryError** – Bad path to json_config
- **JSONDecodeError** – Bad path to json_config

4.1.3 tdameritrade_cli.tools.ods_writer module

class tdameritrade_cli.tools.ods_writer.**ODSWriter**(*sheet_path: str*)

Bases: object

static new_position(*position: List[str]*) → collections.OrderedDict

Create a sheet with a new position.

Parameters **position** – List of the form [position_id, position_type, position_value] to create.

Returns The new sheet with the first data point given by position included.

sheet_data

The data currently in the sheet located at self.sheet_path.

Returns OrderedDict of data.

update_position(*position: List[str], pos_sheet: List[List[T]]*) → List[List[T]]
Update an existing position in an ods document

Parameters

- **position** – List of the form [position_id, position_type, position_value] to update.
- **pos_sheet** – The sheet in the ods document describing the existing position.

Returns Updated pos_sheet

write_positions(*liq_value: str, positions: List[List[str]]*)
Write positions data to an ods document.

Parameters

- **liq_value** – The current liquidation value of a TDA portfolio.
- **positions** – *List[List[str]]* where each *List[str]* is *[position_id, position_type, position_value]*.

tdameritrade_cli.tools.ods_writer.**check_version**(*func: Callable*) → Callable
Context manager that checks the written spreadsheet has a supported version.

4.1.4 tdameritrade_cli.tools.positions module

tdameritrade_cli.tools.positions.**get_positions**(*client: tdameritrade_client.client.TDClient*) → Tuple[str, List[List[str]]]
Retrieve all positions for the authenticated account.

Parameters **client** – An authorized TDClient object.

Returns The current liquidation value of the account and a list of positions where each position is *[position_id, position_type, position_value]*.

4.1.5 Module contents

CHAPTER

FIVE

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

t

INDEX