
Stockfighter Documentation

Release 0.1.0

Scott Triglia

January 04, 2016

1	Stockfighter	1
1.1	Get things started	1
1.2	Features	1
2	Other links	3
2.1	Contributing	3
2.2	Credits	5
2.3	History	5
3	Indices and tables	7

Stockfighter

API wrapper for Stockfighter

- Free software: ISC license
- Limited documentation exists at: <https://stockfighter.readthedocs.org>. Happily accepting patches to improve it!

1.1 Get things started

Not hard!

```
pip install stockfighter
```

Using the standard API

```
from stockfighter import Stockfighter
s = Stockfighter(venue='TESTEX', account='EXB123456', api_key='123456')
print(s.venue_stocks())
```

...and the GM API for managing levels and such

```
from stockfighter import GM
gm = GM(api_key=123456)
print(gm.start('first_steps')) # Start the first level programmatically
```

For simplicity (and so you can share your code!) the GM and Stockfighter clients will also default to reading api keys from the environment variable `API_KEY`, if none was passed.

1.2 Features

- Calling the core Stockfighter API is pretty important :)
- Includes some rudimentary support for the GM API, such as it is known
-get back to me later on what else

2.1 Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given. You can contribute in many ways:

2.1.1 Types of Contributions

Report Bugs

Report bugs at <https://github.com/striglia/stockfighter/issues>.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” is open to whoever wants to implement it.

Implement Features

Look through the GitHub issues for features. Anything tagged with “feature” is open to whoever wants to implement it.

Write Documentation

Stockfighter could always use more documentation, whether as part of the official Stockfighter docs, in docstrings, or even on the web in blog posts, articles, and such.

Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/striglia/stockfighter/issues>.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

2.1.2 Get Started!

Ready to contribute? Here's how to set up *stockfighter* for local development.

1. Fork the *stockfighter* repo on GitHub.
2. Clone your fork locally:

```
$ git clone git@github.com:your_name_here/stockfighter.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv stockfighter
$ cd stockfighter/
$ python setup.py develop
```

4. Create a branch for local development:

```
$ git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

5. When you're done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 stockfighter tests
$ python setup.py test
$ tox
```

To get flake8 and tox, just pip install them into your virtualenv.

6. Commit your changes and push your branch to GitHub:

```
$ git add .
$ git commit -m "Your detailed description of your changes."
$ git push origin name-of-your-bugfix-or-feature
```

7. Submit a pull request through the GitHub website.

2.1.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.

3. The pull request should work for Python 2.6, 2.7, 3.3, 3.4 and 3.5, and for PyPy. Check https://travis-ci.org/striglia/stockfighter/pull_requests and make sure that the tests pass for all supported Python versions.

2.1.4 Tips

To run a subset of tests:

```
$ python -m unittest tests.test_stockfighter
```

2.2 Credits

2.2.1 Development Lead

- Scott Triglia <scott.triglia@gmail.com>

2.2.2 Contributors

- Gergely Imreh (imrehg@gmail.com)
- Chris Hepner (chrishepner)
- Zhen Wang (nehz)
- Kornelijus Survila (kornholi)

2.3 History

2.3.1 0.2.8 (in-progress)

- Set proper min version on requests in setup.py
- Use sessions from requests for Stockfighter API client

2.3.2 0.2.7 (2015-12-21)

- Fix status_for_all_orders_in_a_stock URL
- Fix GM restart/resume/etc failing due to broken format strings

2.3.3 0.2.6 (2015-12-14)

- Fix place_new_order to actually pass price. Novel!

2.3.4 0.2.5 (2015-12-14)

- Python3 re-compatibility
- Fix status_for_all_orders endpoint

2.3.5 0.2.4 (2015-12-12)

- Fix some outlying GM implementation bugs
- Add test for basic GM client object instantiation

2.3.6 0.2.3 (2015-12-12)

- Tests now also assert the API is responding w/ good “ok” values, not just HTTP 200s.
- Py 3.5 testing included

2.3.7 0.2.2 (2015-12-12)

- Fix testing and implementation for place_order

2.3.8 0.2.2 (2015-12-11)

- Make py2 and py3 compatibility a thing, guaranteed by Tox and Travis.

2.3.9 0.2.1 (2015-12-11)

- Working README and quick install

2.3.10 0.2.0 (2015-12-11)

- All API functions implemented and tested

2.3.11 0.1.0 (2015-12-11)

- First release on PyPI.

Indices and tables

- `genindex`
- `modindex`
- `search`