
twstock Documentation

1.0.1

Louie Lu

2019 05 15

Contents

1	twstock -	1
1.1	- User's Guide	1
1.2	API - API Reference	7
1.3	- Additional Notes	12
2	Indices and tables	13
	Python	15

CHAPTER 1

twstock -

twstock twstock

twstock Python 3 Python 3.6.2 Python 2.7 Python 3

issues twstock [gitter.im](#)

1.1 - User's Guide

1.1.1

: twstock Python 3

Python 3 Python 2

twstock (pip)

twstock pip :

```
$ pip install --user twstock
```

sudo:

```
$ sudo pip install twstock
```

twstock

twstock [github](#) git clone [github](#) git clone:

```
$ git clone https://github.com/mlouielu/twstock
$ cd twstock
$ ls
docs  flit.ini  LICENSE  MANIFEST.in  README.md  requirements.txt
setup.py  test  twstock
```

twstock ()

twstock flit pip:

```
$ pip install flit
```

flit *twstock*:

```
$ flit install
```

1.1.2

Python REPL *twstock*

TPEX / TWSE Codes

twstock TPEX/TWSE Codes

:

CLI:

```
$ twstock -U
```

Python:

```
>>> import twstock
>>> twstock.__update_codes()
```

Stock

twstock *Stock*

:

```
>>> import twstock
>>> stock = twstock.Stock('2330')
>>> stock.sid #
'2330'
>>> stock.price #
[207.5, 208.0, 207.0, 208.0, 211.5, 213.0, 216.5, 215.5, 218.0,
 217.0, 215.0, 211.5, 208.5, 210.0, 208.5, 209.0, 207.0, 208.5,
 207.5, 206.0, 206.0, 212.0, 210.5, 214.5, 213.0, 213.0, 214.0,
```

(continues on next page)

()

```

214.5, 215.5, 214.0, 214.5]
>>> stock.high #
[210.0, 208.5, 209.5, 208.0, 212.0, 213.0, 218.0, 217.0, 218.0,
 218.5, 215.0, 214.0, 210.0, 210.5, 208.5, 209.0, 208.5, 208.5,
 208.5, 207.5, 207.0, 212.0, 212.5, 216.0, 214.5, 215.5, 214.0,
 215.0, 215.5, 215.0, 214.5]

```

Stock date :

```

>>> stock.date #
[datetime.datetime(2017, 6, 12, 0, 0),
 datetime.datetime(2017, 6, 13, 0, 0),
 datetime.datetime(2017, 6, 14, 0, 0),
 datetime.datetime(2017, 6, 15, 0, 0),
 ...,
 datetime.datetime(2017, 7, 21, 0, 0),
 datetime.datetime(2017, 7, 24, 0, 0)]

```

Stock 31 fetch :

```

>>> stock.fetch(2015, 7) # 2015 7
>>> stock.fetch(2010, 5) # 2010 5
>>> stock.fetch_31() # 31
>>> stock.fetch_from(2000, 10) # 2000 10

```

Stock method :

```

>>> stock.moving_average(stock.price, 5) #
[208.4, 209.5, 211.2, 212.9, 214.9, 216.0, 216.4, 215.4,
 214.0, 212.4, 210.7, 209.5, 208.6, 208.6, 208.1, 207.6,
 207.0, 208.0, 208.4, 209.8, 211.2, 212.6, 213.0, 213.8,
 214.0, 214.2, 214.5]
>>> stock.moving_average(stock.capacity, 5) #
[40904388.2, 31779953.2, 27540112.6, 28800229.2, 30121867.6,
 31487778.6, 40018023.8, 43162160.8, 44540048.6, 44730965.6,
 43135743.0, 35320904.6, 30738402.0, 24976223.8, 22618522.2,
 20590067.6, 19042051.8, 21642392.4, 22327332.0, 29302556.6,
 29461849.0, 31076569.0, 27909064.6, 26663795.2, 20795579.4,
 19407173.2, 19127688.4]
>>> stock.ma_bias_ratio(5, 10) #
[3.8000000000000114, 3.450000000000017, 2.0999999999999943,
 0.5500000000000114, -1.25, -2.6500000000000057,
 -3.4499999999999886, -3.4000000000000057, -2.700000000000017,
 -2.1500000000000057, -1.5500000000000114, -1.25,
 -0.30000000000001137, -0.09999999999999432,
 0.8500000000000227, 1.799999999999983, 2.799999999999983, 2.5,
 2.700000000000017, 2.0999999999999943, 1.5, 0.9499999999999886]

```

BestFourPoint

BestFourPoint toomore/grs

- /
- /
- /
- /

:

```
>>> stock = twstock.Stock('2330')
>>> bfp = twstock.BestFourPoint(stock)
>>> bfp.best_four_point_to_buy() #
', '
>>> bfp.best_four_point_to_sell() #
False
>>> bfp.best_four_point() #
(True, ', ')
```

: BestFourPoint Stock wrapper Stock BestFourPoint

realtime

realtime *realtime.mock*

realtime (2330):

```
>>> import twstock
>>> stock = twstock.realtime.get('2330') #
{
  "timestamp": 1500877800.0,
  "info": {
    "code": "2330",
    "channel": "2330.tw",
    "name": "",
    "fullname": "",
    "time": "2017-07-24 14:30:00"
  },
  "realtime": {
    "latest_trade_price": "214.50",
    "trade_volume": "4437",
    "accumulate_trade_volume": "19955",
    "best_bid_price": [
      "214.00",
      "213.50",
      "213.00",
      "212.50",
      "212.00"
    ],
  },
}
```

(continues on next page)

()

```

    "best_bid_volume": [
        "29",
        "1621",
        "2056",
        "1337",
        "1673"
    ],
    "best_ask_price": [
        "214.50",
        "215.00",
        "215.50",
        "216.00",
        "216.50"
    ],
    "best_ask_volume": [
        "736",
        "3116",
        "995",
        "1065",
        "684"
    ],
    "open": "213.50",
    "high": "214.50",
    "low": "213.00"
},
"success": true
}
>>> stock = twstock.realtime.get('6223') #
>>> stock
{'timestamp': 1500877800.0, 'info': {'code': '6223', 'channel': '6223.tw',
'name': '', 'fullname': '', 'time': '2017-07-24 14:30:00'},
'realtime': ..., 'success': True}

```

success

realtime success True False rtmessage rtcode:

```

>>> stock = twstock.realtime.get('2330')
>>> stock['success']
True
>>> stock = twstock.realtime.get('')
>>> stock['success']
False
>>> stock
{'rtmessage': 'Information Data Not Found.', 'rtcode': '9999',
'success': False}
>>> stock = twstock.realtime.get('9999')
>>> stock['success']
False
>>> stock
{'msgArray': [], 'userDelay': 0, 'rtmessage': 'Empty Query.',
'referer': '', 'queryTime': {'sysTime': '17:27:02',
'sessionLatestTime': -1, 'sysDate': '20170724', 'sessionKey':
'tse_9999.tw_20170724|', 'sessionFromTime': -1, 'stockInfoItem': 1719,
'showChart': False, 'sessionStr': 'UserSession', 'stockInfo': 277019},

```

(continues on next page)

)

```
'rtcode': '5001', 'success': False}
```

realtime:

```
>>> stocks = twstock.realtime.get(['2330', '2337', '2409'])
>>> stocks['success']
>>> stocks
{'2330': {'timestamp': 1500877800.0, ..., 'success': True},
 '2337': {'timestamp': 1500877800.0, ..., 'success': True},
 '2409': {'timestamp': 1500877800.0, ..., 'success': True},
 'success': True}
>>> stocks['2330']['success']
True
```

mock

```
>>> twstock.realtime.mock = True
>>> twstock.realtime.get('2337')
```

Codes

codes codes.tpexcodes.twsecodes.codes

:

```
>>> import twstock
>>> '2330' in twstock.twse
True
>>> '6223' in twstock.twse
False
```

:

```
>>> '2330' in twstock.tpex
False
>>> '6223' in twstock.tpex
True
```

:

```
>>> '2330' in twstock.codes
True
>>> '6223' in twstock.codes
True
```

Legacy

Legacy toomore/grs grs LegacyAnalytics LegacyBestFourPoint

CLI tools

twstock command line tools :

```
$ twstock -s 2330 6223
----- 2330 -----
high : 215.0 214.0 210.0 210.5 208.5
low  : 212.0 211.0 208.0 208.5 206.5
price: 215.0 211.5 208.5 210.0 208.5
----- 2337 -----
high : 16.2 16.8 16.4 16.75 16.75
low  : 15.8 16.1 15.15 16.3 16.25
price: 15.95 16.25 16.25 16.6 16.7

$ twstock -b 2330
Best Four Point
-----
2330: Buy
6223: Sell ,
```

1.2 API - API Reference

class method

1.2.1 API Reference

stock —

stock *DATATUPLE* namedtuple *BaseFetcher* *TWSEFetcher* *TPEXFetcher* class *Stock* *Stock*
fetcher

DATATUPLE

class stock.**DATATUPLE** (*date, capacity, turnover, open, high, low, close, change, transaction*)
nametuple

Attributes:

date

datetime.datetime datetime.datetime(2017, 6, 12, 0, 0)

capacity

(:)

turnover

(:/)

open

high

low

close

change

transaction

Stock

```
class stock.Stock(sid: str, initial_fetch: bool=True)
    (...etc) Stock initial_fetch True() fetch_31() 31
```

Class attributes are:

sid

fetcher

instance fetcher

raw_data

TWSEFetcher TPEXFetcher

data

raw_data DATATUPLE

Fetcher method:

fetch (*self*, year: int, month: int)

fetch_from (*self*, year: int, month: int)

fetch_31 (*self*)

31

method:

continuous (*self*, data)

data

moving_average (*self*, data: list, days: int)

data days

ma_bias_ratio (*self*, day1, day2)

day1 day2

ma_bias_ratio_pivot (*self*, data, sample_size=5, position=False)

Fetcher

```
class stock.BaseFetcher
```

fetch (*self*, year, month, sid, retry)

_convert_date (*self*, date)

:

```
>>> date = self._convert_date('106/05/01')
>>> print(date)
'2017/05/01'
```

_make_datatuple (*self*, data)

DATATUPLE

purify (*self*, original_data: list)

original_data *DATATUPLE*

```
class stock.TWSEFetcher(BaseFetcher)
```

```
class stock.TPEXFetcher(BaseFetcher)
```

analytics –

```
analytics Analytics BestFourPoint Analytics stock.Stock stock.Stock
```

```
: analytics stock.Stock realtime
```

- Analytics

```
class analytics.Analytics
```

continuous (d

```
    data(list)–
```

```
    data
```

```
    moving_average(data, days)
```

```
    • data(list)–
```

```
    • days(int)–
```

```
    data days:
```

```
>>> stock.moving_average(stock.price, 5)      # 5
>>> stock.moving_average(stock.capacity, 5)   # 5
```

```
    ma_bias_ratio(day1, day2)
```

```
    • day1(int)– n
```

```
    • day2(int)– m
```

```
    (), day1 - day2
```

```
    ma_bias_ratio_pivot(data, sample_size=5, positive=False)
```

```
    • data(list)–, price
```

```
    • sample_size(int)–
```

```
    • positive(bool)– (True), (False)
```

- BestFourPoint

```
class analytics.BestFourPoint(stock)
```

```
    stock(stock.Stock)–
```

```
    bias_ratio(position=False)
```

```
        positive(bool) - (True), (False)
3,6
plus_bias_ratio()
3,6
mins_bias_ratio()
3,6
best_buy_1()
    bool
:
best_buy_2()
    bool
:
best_buy_3()
    bool
:
best_buy_4()
    bool
:
best_sell_1()
    bool
:
best_sell_2()
    bool
:
best_sell_3()
    bool
:
best_sell_4()
    bool
:
best_four_point_to_buy()
    bool, str
    False
best_four_point_to_sell()
    bool, str
    False
best_four_point()
```

```
(bool, str), None
(True, msg) (False, msg) None
```

realtime —

realtime API *get()*

Attributes:

`realtime.mock`

mock:

```
>>> twstock.realtime.mock = True #
>>> twstock.realtime.mock = False #
```

Methods:

`realtime.get(stocks)`

stocks (str or list[str]) – list:

```
>>> realtime.get('2330') #
>>> realtime.get(['2330', '6223']) #
```

dict –

```
:
>>> realtime.get('2330')
{
  'timestamp': 1500877800.0,
  'info': {
    'code': '2330',
    'channel': '2330.tw'
    'name': '',
    'fullname': '',
    'time': '2017-07-24 14:30:00'
  },
  'realtime': {
    'latest_trade_price': '214.50',
    'trade_volume': '4437',
    'accumulate_trade_volume': '19955',
    'best_bid_price': ['214.00', '213.50', '213.00', '212.50', '212.00'],
    'best_bid_volume': ['29', '1621', '2056', '1337', '1673'],
    'best_ask_price': ['214.50', '215.00', '215.50', '216.00', '216.50'],
    'best_ask_volume': ['736', '3116', '995', '1065', '684'],
    'open': '213.50',
    'high': '214.50',
    'low': '213.00'
  },
  'success': True
}
```

```
:
>>> realtime.get(['2330', '2337'])
{
  '2330': ...,
```

(continues on next page)

()

```
'2337': ...,  
'success': True  
}
```

codes –

`codes` `twstock.codestwstock.tpextwstock.twse`

class `codes.StockCodeInfo` (*type, code, name, ISIN, start, market, group, CFI*)

`codes.codes`

`codes.tpex`

`codes.twse`

1.3 - Additional Notes

1.3.1 - Contributing

twstock

bug [GitHub issues](#) issue patch [GitHub pull request](#) patch patch [Contributors List](#)

Coding Style Guide

Coding Style [PEP 8](#)

1.3.2 LICENSE

Copyright (c) 2017-2019 Louie Lu

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the Software), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED AS IS, WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

CHAPTER 2

Indices and tables

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

a

analytics, [9](#)

c

codes, [12](#)

r

realtime, [11](#)

s

stock, [7](#)