
finvizfinance

Release 0.8

Feb 05, 2021

Contents

1	Installation (python >= v3.5)	3
2	Quote Examples	5
3	Screener Example	7
4	News Example	9
5	Insider Example	11
6	Contents	13
6.1	Quote	13
6.2	Screener	14
6.3	Group	16
6.4	News	17
6.5	Insider	18
6.6	Forex	18
6.7	Crypto	18
6.8	Future	19
6.9	Earnings	19
7	Indices and tables	21
	Python Module Index	23
	Index	25

It is a Finviz Finance information downloader.

CHAPTER 1

Installation (python >= v3.5)

```
> virtualenv -p python3 virtualenvironment  
> source virtualenvironment/bin/activate  
> pip install finvizfinance
```


CHAPTER 2

Quote Examples

```
import pandas as pd
from finvizfinance.quote import finvizfinance

stock = finvizfinance('tsla')
```

Example downloading chart:

```
stock.TickerCharts(out_dir='asset')
```

Example getting individual ticker information

```
stock_fundament = stock.TickerFundament()
stock_description = stock.TickerDescription()
outer_ratings_df = stock.TickerOuterRatings()
news_df = stock.TickerNews()
inside_trader_df = stock.TickerInsideTrader()
```


CHAPTER 3

Screeener Example

```
from finvizfinance.screener.overview import Overview
foverview = Overview()
filters_dict = {'Exchange': 'AMEX', 'Sector': 'Basic Materials'}
foverview.set_filter(filters_dict=filters_dict)
df = foverview.ScreenerView()
df.head()
```


CHAPTER 4

News Example

```
from finvizfinance.news import News
fnews = News()
all_news = fnews.getNews()
# all_news['news'].head()
# all_news['blogs'].head()
```


CHAPTER 5

Insider Example

```
from finvizfinance.insider import Insider
finsider = Insider(option='top owner trade')
finsider.getInsider().head()
```


6.1 Quote

Getting individual ticker information.

class `finvizfinance.quote.Quote`
quote Getting current price of the ticker

getCurrent (*ticker*)
Getting current price of the ticker.

Returns price of the ticker

Return type price(float)

class `finvizfinance.quote.finvizfinance` (*ticker, verbose=0*)
Getting information from the individual ticker.

Parameters

- **ticker** (*str*) – ticker string
- **verbose** (*int*) – choice of visual the progress. 1 for visualize progress.

TickerCharts (*timeframe='daily', charttype='advanced', out_dir="", urlonly=False*)
Download ticker charts.

Parameters

- **timeframe** (*str*) – choice of timeframe (daily, weekly, monthly).
- **charttype** (*str*) – choice of type of chart (candle, line, advanced).
- **out_dir** (*str*) – output image directory. default none.
- **urlonly** (*bool*) – choice of downloading charts, default: downloading chart

Returns url for the chart

Return type charturl(str)

TickerDescription()
Get ticker description.
Returns ticker description.
Return type description(str)

TickerFullInfo()
Get all the ticker information.
Returns insider information table
Return type df(pandas.DataFrame)

TickerFundament()
Get ticker fundament.
Returns ticker fundament.
Return type fundament(dict)

TickerInsideTrader()
Get insider information table.
Returns insider information table
Return type df(pandas.DataFrame)

TickerNews()
Get news information table.
Returns news information table
Return type df(pandas.DataFrame)

TickerOuterRatings()
Get outer ratings table.
Returns outer ratings table
Return type df(pandas.DataFrame)

TickerSignal()
Get all the trading signals from finviz.
Returns get all the ticker signals as list.
Return type ticker_signals(list)

6.2 Screener

Getting screener information.

6.2.1 Overview

class finvizfinance.screener.overview.**Overview**
Getting information from the finviz screener overview page.

ScreenerView (order='ticker', limit=-1, verbose=1, ascend=True)
Get screener table.

Parameters

- **order** (*str*) – sort the table by the choice of order.
- **limit** (*int*) – set the top k rows of the screener.
- **verbose** (*int*) – choice of visual the progress. 1 for visualize progress.
- **ascend** (*bool*) – if True, the order is ascending.

Returns screener information table

Return type df(pandas.DataFrame)

compare (*ticker, compare_list, order='ticker', verbose=1*)

Get screener table of similar property (Sector, Industry, Country)

Parameters

- **ticker** (*str*) – the ticker to compare
- **compare_list** (*list*) – choice of compare property (Sector, Industry, Country) or combination.
- **order** (*str*) – sort the table by the choice of order
- **verbose** (*int*) – choice of visual the progress. 1 for visualize progress

Returns screener information table

Return type df(pandas.DataFrame)

getFilterOptions (*screen_filter*)

Get filters options.

Parameters **screen_filter** (*str*) – screen filter for checking options

Returns all the available filters

Return type filter_options(list)

getFilters ()

Get filters.

Returns all the available filters

Return type filters(list)

getOrders ()

Get orders.

Returns all the available orders

Return type orders(list)

getSignal ()

Get signals.

Returns all the available trading signals

Return type signals(list)

set_filter (*signal=", filters_dict={}, ticker="*)

Update the settings.

Parameters

- **signal** (*str*) – ticker signal
- **filters_dict** (*dict*) – dictionary of filters

- **ticker** (*str*) – ticker string

6.2.2 Valuation

class finvizfinance.screener.valuation.Valuation

Valuation inherit from overview module. Getting information from the finviz screener valuation page.

6.2.3 Financial

class finvizfinance.screener.financial.Financial

Financial inherit from overview module. Getting information from the finviz screener financial page.

6.2.4 Ownership

class finvizfinance.screener.ownership.Ownership

Ownership inherit from overview module. Getting information from the finviz screener ownership page.

6.2.5 Technical

class finvizfinance.screener.technical.Technical

Technical inherit from overview module. Getting information from the finviz screener technical page.

6.2.6 Tickers

class finvizfinance.screener.ticker.Ticker

Financial inherit from overview module. Getting information from the finviz screener ticker page.

ScreenerView (*limit=-1, verbose=1*)

Get screener table.

Parameters **verbose** (*int*) – choice of visual the progress. 1 for visualize progress.

Returns get all the tickers as list.

Return type tickers(list)

6.3 Group

Getting group information.

6.3.1 Overview

class finvizfinance.group.overview.Overview

Getting information from the finviz group overview page.

ScreenerView (*group='Sector', order='Name'*)

Get screener table.

Parameters

- **group** (*str*) – choice of group option.

- **order** (*str*) – sort the table by the choice of order.

Returns group information table.

Return type df(pandas.DataFrame)

getGroup ()

Get groups.

Returns all the available groups.

Return type groups(list)

getOrders ()

Get orders.

Returns all the available orders.

Return type orders(list)

6.3.2 Valuation

class finvizfinance.group.valuation.Valuation

Valuation inherit from overview module. Getting information from the finviz group valuation page.

6.3.3 Performance

class finvizfinance.group.performance.Performance

Performance inherit from overview module. Getting information from the finviz group performance page.

6.3.4 Spectrum

class finvizfinance.group.spectrum.Spectrum

Spectrum inherit from overview module. Getting information from the finviz group spectrum page.

ScreenableView (*group='Sector', order='Name', out_dir=""*)

Get screener table.

Parameters

- **group** (*str*) – choice of group option.
- **order** (*str*) – sort the table by the choice of order.

6.4 News

Getting trading news.

class finvizfinance.news.News

Getting information from the finviz news page.

getNews ()

Get insider information table.

Retrieves table information from finviz finance news.

Returns news table

Return type news(dict)

6.5 Insider

Getting trading insider.

class finvizfinance.insider.**Insider** (*option='latest'*)

Getting information from the finviz insider page.

Parameters **option** (*str*) – choose a option (latest, top week, top owner trade, insider_id)

getInsider ()

Get insider information table.

Returns insider information table

Return type df(pandas.DataFrame)

6.6 Forex

Getting foreign exchange information.

class finvizfinance.forex.**Forex**

Getting information from the finviz forex page.

chart (*forex, timeframe='D', urlonly=False*)

Get forex chart.

Parameters

- **forex** (*str*) – foreign exchange name
- **timeframe** (*str*) – choice of timeframe(5M, H, D, W, M)
- **urlonly** (*bool*) – choice of downloading charts, default: downloading chart

Returns url for the chart

Return type charturl(str)

performance (*change='percent'*)

Get forex performance table.

Parameters **change** (*str*) – choose an option of change(percent, PIPS)

Returns forex performance table

Return type df(pandas.DataFrame)

6.7 Crypto

Getting crypto currency information.

class finvizfinance.crypto.**Crypto**

Getting information from the finviz crypto page.

chart (*crypto, timeframe='D', urlonly=False*)

Get crypto chart.

Parameters

- **crypto** (*str*) – crypto currency
- **timeframe** (*str*) – choice of timeframe(5M, H, D, W, M)
- **urlonly** (*bool*) – choice of downloading charts, default: downloading chart

Returns url for the chart**Return type** charturl(*str*)**performance** ()

Get crypto performance table.

Returns crypto performance table**Return type** df(*pandas.DataFrame*)

6.8 Future

Getting future information.

class finvizfinance.future.Future

Getting information from the finviz future page.

performance (*timeframe='D'*)

Get forex performance table.

Parameters **timeframe** (*str*) – choice of timeframe(D, W, M, Q, HY, Y)**Returns** forex performance table**Return type** df(*pandas.DataFrame*)

6.9 Earnings

Getting earning dates information.

class finvizfinance.earnings.Earnings (*period='This Week'*)

Partition dataframe of ticker information of period of earning dates(This Week, Next Week, Previous Week, This Month) into dates

Parameters **period** (*str*) – choose an option of period(This Week, Next Week, Previous Week, This Month).**outputCSV** (*output_dir='earning_days'*)

Output dataframes to csv files.

Parameters **output_dir** (*str*) – name of the output directory.**outputExcel** (*output_file='earning_days.xlsx'*)

Output dataframes to single Excel file.

Parameters **output_file** (*str*) – name of the output excel file.**partitionDays** (*mode='financial'*)

Partition dataframe to separate dataframes according to the dates.

Parameters **mode** (*str*) – choose an option of period(financial, overview, valuation, ownership, performance, technical).

CHAPTER 7

Indices and tables

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

f

- `finvizfinance.crypto`, 18
- `finvizfinance.earnings`, 19
- `finvizfinance.forex`, 18
- `finvizfinance.future`, 19
- `finvizfinance.group.overview`, 16
- `finvizfinance.group.performance`, 17
- `finvizfinance.group.spectrum`, 17
- `finvizfinance.group.valuation`, 17
- `finvizfinance.insider`, 18
- `finvizfinance.news`, 17
- `finvizfinance.quote`, 13
- `finvizfinance.screener.financial`, 16
- `finvizfinance.screener.overview`, 14
- `finvizfinance.screener.ownership`, 16
- `finvizfinance.screener.technical`, 16
- `finvizfinance.screener.ticker`, 16
- `finvizfinance.screener.valuation`, 16

C

`chart()` (*finvizfinance.crypto.Crypto method*), 18
`chart()` (*finvizfinance.forex.Forex method*), 18
`compare()` (*finvizfinance.screener.overview.Overview method*), 15
`Crypto` (*class in finvizfinance.crypto*), 18

E

`Earnings` (*class in finvizfinance.earnings*), 19

F

`Financial` (*class in finvizfinance.screener.financial*), 16
`finvizfinance` (*class in finvizfinance.quote*), 13
`finvizfinance.crypto` (*module*), 18
`finvizfinance.earnings` (*module*), 19
`finvizfinance.forex` (*module*), 18
`finvizfinance.future` (*module*), 19
`finvizfinance.group.overview` (*module*), 16
`finvizfinance.group.performance` (*module*), 17
`finvizfinance.group.spectrum` (*module*), 17
`finvizfinance.group.valuation` (*module*), 17
`finvizfinance.insider` (*module*), 18
`finvizfinance.news` (*module*), 17
`finvizfinance.quote` (*module*), 13
`finvizfinance.screener.financial` (*module*), 16
`finvizfinance.screener.overview` (*module*), 14
`finvizfinance.screener.ownership` (*module*), 16
`finvizfinance.screener.technical` (*module*), 16
`finvizfinance.screener.ticker` (*module*), 16
`finvizfinance.screener.valuation` (*module*), 16
`Forex` (*class in finvizfinance.forex*), 18
`Future` (*class in finvizfinance.future*), 19

G

`getCurrent()` (*finvizfinance.quote.Quote method*), 13
`getFilterOptions()` (*finvizfinance.screener.overview.Overview method*), 15
`getFilters()` (*finvizfinance.screener.overview.Overview method*), 15
`getGroup()` (*finvizfinance.group.overview.Overview method*), 17
`getInsider()` (*finvizfinance.insider.Insider method*), 18
`getNews()` (*finvizfinance.news.News method*), 17
`getOrders()` (*finvizfinance.group.overview.Overview method*), 17
`getOrders()` (*finvizfinance.screener.overview.Overview method*), 15
`getSignal()` (*finvizfinance.screener.overview.Overview method*), 15

I

`Insider` (*class in finvizfinance.insider*), 18

N

`News` (*class in finvizfinance.news*), 17

O

`outputCSV()` (*finvizfinance.earnings.Earnings method*), 19
`outputExcel()` (*finvizfinance.earnings.Earnings method*), 19
`Overview` (*class in finvizfinance.group.overview*), 16
`Overview` (*class in finvizfinance.screener.overview*), 14
`Ownership` (*class in finvizfinance.screener.ownership*), 16

P

`partitionDays()` (*finvizfinance.earnings.Earnings method*), 19

`Performance` (*class in finvizfinance.group.performance*), 17

`performance()` (*finvizfinance.crypto.Crypto method*), 19

`performance()` (*finvizfinance.forex.Forex method*), 18

`performance()` (*finvizfinance.future.Future method*), 19

Q

`Quote` (*class in finvizfinance.quote*), 13

S

`ScreenableView()` (*finvizfinance.group.overview.Overview method*), 16

`ScreenableView()` (*finvizfinance.group.spectrum.Spectrum method*), 17

`ScreenableView()` (*finvizfinance.screener.overview.Overview method*), 14

`ScreenableView()` (*finvizfinance.screener.ticker.Ticker method*), 16

`set_filter()` (*finvizfinance.screener.overview.Overview method*), 15

`Spectrum` (*class in finvizfinance.group.spectrum*), 17

T

`Technical` (*class in finvizfinance.screener.technical*), 16

`Ticker` (*class in finvizfinance.screener.ticker*), 16

`TickerCharts()` (*finvizfinance.quote.finvizfinance method*), 13

`TickerDescription()` (*finvizfinance.quote.finvizfinance method*), 13

`TickerFullInfo()` (*finvizfinance.quote.finvizfinance method*), 14

`TickerFundament()` (*finvizfinance.quote.finvizfinance method*), 14

`TickerInsideTrader()` (*finvizfinance.quote.finvizfinance method*), 14

`TickerNews()` (*finvizfinance.quote.finvizfinance method*), 14

`TickerOuterRatings()` (*finvizfinance.quote.finvizfinance method*), 14

`TickerSignal()` (*finvizfinance.quote.finvizfinance method*), 14

V

`Valuation` (*class in finvizfinance.group.valuation*), 17

`Valuation` (*class in finvizfinance.screener.valuation*), 16