

---

# **finvizfinance**

***Release 0.14.4***

**Oct 18, 2022**



---

## Contents

---

|          |   |           |
|----------|---|-----------|
| <b>1</b> | <b>Installation (python &gt;= v3.5)</b> | <b>3</b>  |
| <b>2</b> | <b>Quote Examples</b>                   | <b>5</b>  |
| <b>3</b> | <b>Screener Example</b>                 | <b>7</b>  |
| <b>4</b> | <b>News Example</b>                     | <b>9</b>  |
| <b>5</b> | <b>Insider Example</b>                  | <b>11</b> |
| <b>6</b> | <b>Contents</b>                         | <b>13</b> |
| 6.1      | Quote . . . . .                         | 13        |
| 6.2      | Screener . . . . .                      | 15        |
| 6.3      | Group . . . . .                         | 17        |
| 6.4      | News . . . . .                          | 19        |
| 6.5      | Insider . . . . .                       | 19        |
| 6.6      | Forex . . . . .                         | 19        |
| 6.7      | Crypto . . . . .                        | 20        |
| 6.8      | Future . . . . .                        | 20        |
| 6.9      | Earnings . . . . .                      | 20        |
| 6.10     | Calendar . . . . .                      | 21        |
| <b>7</b> | <b>Indices and tables</b>               | <b>23</b> |
|          | <b>Python Module Index</b>              | <b>25</b> |
|          | <b>Index</b>                            | <b>27</b> |



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.ticker_fundament()
stock_description = stock.ticker_description()
outer_ratings_df = stock.ticker_outer_ratings()
news_df = stock.ticker_news()
inside_trader_df = stock.ticker_inside_trader()
```



## 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.screener_view()
df.head()
```



## CHAPTER 4

---

### News Example

---

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



## CHAPTER 5

---

### Insider Example

---

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





## 6.1 Quote

Getting individual ticker information.

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

```
    get_current (ticker)  
        Getting current price of the ticker.
```

**Returns** price of the ticker

**Return type** price(float)

```
class finvizfinance.quote.Statements  
    Getting statements of ticker
```

```
    get_statements (ticker, statement='I', timeframe='A')  
        Getting statements of ticker.
```

**Parameters**

- **ticker** (*str*) – ticker string
- **statement** (*str*) – I(Income Statement), B(Balace Sheet), C(Cash Flow)
- **timeframe** (*str*) – A(Annual), Q(Quarter)

**Returns** statements table

**Return type** df(pandas.DataFrame)

```
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.

**ticker\_charts** (*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)

**ticker\_description** ()  
Get ticker description.

**Returns** ticker description.

**Return type** description(str)

**ticker\_full\_info** ()  
Get all the ticker information.

**Returns** insider information table

**Return type** df(pandas.DataFrame)

**ticker\_fundament** (*raw=True, output\_format='dict'*)  
Get ticker fundament.

**Parameters**

- **raw** (*boolean*) – if True, the data is raw.
- **output\_format** (*str*) – choice of output format (dict, series).

**Returns** ticker fundament.

**Return type** fundament(dict)

**ticker\_inside\_trader** ()  
Get insider information table.

**Returns** insider information table

**Return type** df(pandas.DataFrame)

**ticker\_news** ()  
Get news information table.

**Returns** news information table

**Return type** df(pandas.DataFrame)

**ticker\_outer\_ratings** ()  
Get outer ratings table.

**Returns** outer ratings table

**Return type** df(pandas.DataFrame)

**ticker\_signal** ()  
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.

**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)

**get\_filter\_options** (screen\_filter)

Get filters options.

**Parameters** **screen\_filter** (str) – screen filter for checking options

**Returns** all the available filters

**Return type** filter\_options(list)

**get\_filters** ()

Get filters.

**Returns** all the available filters

**Return type** filters(list)

**get\_orders** ()

Get orders.

**Returns** all the available orders

**Return type** orders(list)

**get\_signal** ()

Get signals.

**Returns** all the available trading signals

**Return type** signals(list)

```
screener_view (order='ticker', limit=-1, select_page=None, verbose=1, ascend=True,  
              sleep_sec=1)
```

Get screener table.

#### Parameters

- **order** (*str*) – sort the table by the choice of order.
- **limit** (*int*) – set the top k rows of the screener.
- **select\_page** (*int*) – set the page of the screener.
- **verbose** (*int*) – choice of visual the progress. 1 for visualize progress.
- **ascend** (*bool*) – if True, the order is ascending.
- **sleep\_sec** (*int*) – sleep seconds for fetching each page.

**Returns** screener information table

**Return type** df(pandas.DataFrame)

```
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 Custom

**class** finvizfinance.screener.custom.**Custom**

Custom inherit from overview module. Getting information from the finviz screener custom page.

**get\_columns** ()

Get information about the columns

**Returns** return the index and column name.

**Return type** columns(dict)

**screener\_view** (order='ticker', limit=-1, select\_page=None, verbose=1, ascend=True, columns=[0, 1, 2, 3, 4, 5, 6, 7, 65, 66, 67], sleep\_sec=1)

Get screener table.

**Parameters**

- **order** (*str*) – sort the table by the choice of order.
- **limit** (*int*) – set the top k rows of the screener.
- **select\_page** (*int*) – set the page of the screener.
- **verbose** (*int*) – choice of visual the progress. 1 for visualize progress.
- **ascend** (*bool*) – if True, the order is ascending.
- **columns** (*list*) – columns of your choice. Default index: 0,1,2,3,4,5,6,7,65,66,67.
- **sleep\_sec** (*int*) – sleep seconds for fetching each page.

**Returns** screener information table

**Return type** df(pandas.DataFrame)

## 6.2.7 Tickers

**class** finvizfinance.screener.ticker.**Ticker**

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

**screener\_view** (order='ticker', limit=-1, verbose=1, ascend=True, sleep\_sec=1)

Get screener stocks.

**Parameters**

- **order** (*str*) – sort the list by the choice of order.
- **limit** (*int*) – set the top k stocks of the screener.
- **verbose** (*int*) – choice of visual the progress. 1 for visualize progress.
- **ascend** (*bool*) – if True, the order is ascending.
- **sleep\_sec** (*int*) – sleep seconds for fetching each page.

**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.

**get\_group** ()

Get groups.

**Returns** all the available groups.

**Return type** groups(list)

**get\_orders** ()

Get orders.

**Returns** all the available orders.

**Return type** orders(list)

**screener\_view** (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)

### 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.

**screener\_view** (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.

    get_news ()
        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, latest buys, latest sales, top week, top week buys,
        top week sales, top owner trade, top owner buys, top owner sales, insider_id)

    get_insider ()
        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).

**output\_csv** (*output\_dir='earning\_days'*)  
Output dataframes to csv files.

**Parameters** **output\_dir** (*str*) – name of the output directory.

**output\_excel** (*output\_file='earning\_days.xlsx'*)  
Output dataframes to single Excel file.

**Parameters** **output\_file** (*str*) – name of the output excel file.

**partition\_days** (*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).

## 6.10 Calendar

Getting economic calendar information.

**class** finvizfinance.calendar.**Calendar**  
Getting information from the finviz calendar page.

**calendar** ()  
Get economic calendar table.

**Returns** economic calendar table

**Return type** df(pandas.DataFrame)



## CHAPTER 7

---

### Indices and tables

---

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



### c

[calendar](#), 21  
[crypto](#), 20

### e

[earnings](#), 20

### f

[finvizfinance](#), 13  
[finvizfinance.calendar](#), 21  
[finvizfinance.crypto](#), 20  
[finvizfinance.earnings](#), 20  
[finvizfinance.forex](#), 19  
[finvizfinance.future](#), 20  
[finvizfinance.group.overview](#), 18  
[finvizfinance.group.performance](#), 18  
[finvizfinance.group.spectrum](#), 18  
[finvizfinance.group.valuation](#), 18  
[finvizfinance.insider](#), 19  
[finvizfinance.news](#), 19  
[finvizfinance.quote](#), 13  
[finvizfinance.screener.custom](#), 17  
[finvizfinance.screener.financial](#), 16  
[finvizfinance.screener.overview](#), 15  
[finvizfinance.screener.ownership](#), 16  
[finvizfinance.screener.technical](#), 16  
[finvizfinance.screener.ticker](#), 17  
[finvizfinance.screener.valuation](#), 16  
[forex](#), 19  
[future](#), 20

### g

[group.overview](#), 18  
[group.performance](#), 18  
[group.spectrum](#), 18  
[group.valuation](#), 18

### i

[insider](#), 19

### n

[news](#), 19

### s

[screen.custom](#), 17  
[screen.financial](#), 16  
[screen.overview](#), 15  
[screen.ticker](#), 17  
[screen.valuation](#), 16



**C**

Calendar (*class in finvizfinance.calendar*), 21  
calendar (*module*), 21  
calendar() (*finvizfinance.calendar.Calendar method*), 21  
chart() (*finvizfinance.crypto.Crypto method*), 20  
chart() (*finvizfinance.forex.Forex method*), 19  
compare() (*finvizfinance.screener.overview.Overview method*), 15  
Crypto (*class in finvizfinance.crypto*), 20  
crypto (*module*), 20  
Custom (*class in finvizfinance.screener.custom*), 17

**E**

Earnings (*class in finvizfinance.earnings*), 20  
earnings (*module*), 20

**F**

Financial (*class in finvizfinance.screener.financial*), 16  
finvizfinance (*class in finvizfinance.quote*), 13  
finvizfinance (*module*), 13  
finvizfinance.calendar (*module*), 21  
finvizfinance.crypto (*module*), 20  
finvizfinance.earnings (*module*), 20  
finvizfinance.forex (*module*), 19  
finvizfinance.future (*module*), 20  
finvizfinance.group.overview (*module*), 18  
finvizfinance.group.performance (*module*), 18  
finvizfinance.group.spectrum (*module*), 18  
finvizfinance.group.valuation (*module*), 18  
finvizfinance.insider (*module*), 19  
finvizfinance.news (*module*), 19  
finvizfinance.quote (*module*), 13  
finvizfinance.screener.custom (*module*), 17  
finvizfinance.screener.financial (*module*), 16

finvizfinance.screener.overview (*module*), 15  
finvizfinance.screener.ownership (*module*), 16  
finvizfinance.screener.technical (*module*), 16  
finvizfinance.screener.ticker (*module*), 17  
finvizfinance.screener.valuation (*module*), 16  
Forex (*class in finvizfinance.forex*), 19  
forex (*module*), 19  
Future (*class in finvizfinance.future*), 20  
future (*module*), 20

**G**

get\_columns() (*finvizfinance.screener.custom.Custom method*), 17  
get\_current() (*finvizfinance.quote.Quote method*), 13  
get\_filter\_options() (*finvizfinance.screener.overview.Overview method*), 15  
get\_filters() (*finvizfinance.screener.overview.Overview method*), 15  
get\_group() (*finvizfinance.group.overview.Overview method*), 18  
get\_insider() (*finvizfinance.insider.Insider method*), 19  
get\_news() (*finvizfinance.news.News method*), 19  
get\_orders() (*finvizfinance.group.overview.Overview method*), 18  
get\_orders() (*finvizfinance.screener.overview.Overview method*), 15  
get\_signal() (*finvizfinance.screener.overview.Overview method*), 15

get\_statements() (*finvizfinance.quote.Statements method*), 13  
group.overview (*module*), 18  
group.performance (*module*), 18  
group.spectrum (*module*), 18  
group.valuation (*module*), 18

## I

Insider (*class in finvizfinance.insider*), 19  
insider (*module*), 19

## N

News (*class in finvizfinance.news*), 19  
news (*module*), 19

## O

output\_csv() (*finvizfinance.earnings.Earnings method*), 21  
output\_excel() (*finvizfinance.earnings.Earnings method*), 21  
Overview (*class in finvizfinance.group.overview*), 18  
Overview (*class in finvizfinance.screener.overview*), 15  
Ownership (*class in finvizfinance.screener.ownership*), 16

## P

partition\_days() (*finvizfinance.earnings.Earnings method*), 21  
Performance (*class in finvizfinance.group.performance*), 18  
performance() (*finvizfinance.crypto.Crypto method*), 20  
performance() (*finvizfinance.forex.Forex method*), 19  
performance() (*finvizfinance.future.Future method*), 20

## Q

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

## S

screen.custom (*module*), 17  
screen.financial (*module*), 16  
screen.overview (*module*), 15  
screen.ticker (*module*), 17  
screen.valuation (*module*), 16  
screener\_view() (*finvizfinance.group.overview.Overview method*), 18  
screener\_view() (*finvizfinance.group.spectrum.Spectrum method*), 18  
screener\_view() (*finvizfinance.screener.custom.Custom method*), 17

screener\_view() (*finvizfinance.screener.overview.Overview method*), 15  
screener\_view() (*finvizfinance.screener.ticker.Ticker method*), 17  
set\_filter() (*finvizfinance.screener.overview.Overview method*), 16  
Spectrum (*class in finvizfinance.group.spectrum*), 18  
Statements (*class in finvizfinance.quote*), 13

## T

Technical (*class in finvizfinance.screener.technical*), 16  
Ticker (*class in finvizfinance.screener.ticker*), 17  
ticker\_charts() (*finvizfinance.quote.finvizfinance method*), 14  
ticker\_description() (*finvizfinance.quote.finvizfinance method*), 14  
ticker\_full\_info() (*finvizfinance.quote.finvizfinance method*), 14  
ticker\_fundament() (*finvizfinance.quote.finvizfinance method*), 14  
ticker\_inside\_trader() (*finvizfinance.quote.finvizfinance method*), 14  
ticker\_news() (*finvizfinance.quote.finvizfinance method*), 14  
ticker\_outer\_ratings() (*finvizfinance.quote.finvizfinance method*), 14  
ticker\_signal() (*finvizfinance.quote.finvizfinance method*), 14

## V

Valuation (*class in finvizfinance.group.valuation*), 18  
Valuation (*class in finvizfinance.screener.valuation*), 16