

---

# **finvizfinance**

***Release 0.6.2***

**Aug 21, 2020**



---

## 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 . . . . .	14
6.3	Group . . . . .	15
6.4	News . . . . .	16
6.5	Insider . . . . .	16
6.6	Forex . . . . .	17
6.7	Crypto . . . . .	17
6.8	Earnings . . . . .	17
<b>7</b>	<b>Indices and tables</b>	<b>19</b>
	<b>Python Module Index</b>	<b>21</b>
	<b>Index</b>	<b>23</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.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.finvizfinance` (*ticker*, *verbose=0*)

Getting information from the individual ticker.

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

**TickerCharts** (*timeframe='daily'*, *charttype='advanced'*, *out\_dir=""*)

Download ticker charts.

**Args:** *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.

**TickerDescription** ()

Get ticker description.

**Returns:** *description*(str): ticker description.

**TickerFullInfo** ()

Get all the ticker information.

**Returns:** *df*(pandas.DataFrame): insider information table

**TickerFundament** ()

Get ticker fundament.

**Returns:** *fundament*(dict): ticker fundament.

**TickerInsideTrader** ()

Get insider information table.

**Returns:** *df*(pandas.DataFrame): insider information table

**TickerNews** ()

Get news information table.

**Returns:** df(pandas.DataFrame): news information table

**TickerOuterRatings** ()

Get outer ratings table.

**Returns:** df(pandas.DataFrame): outer ratings table

## 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', verbose=1*)

Get screener table.

**Args:** order(str): sort the table by the choice of order verbose(int): choice of visual the progress. 1 for visualize progress

**Returns:** df(pandas.DataFrame): screener information table

**compare** (*ticker, compare\_list, order='ticker', verbose=1*)

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

**Args:** 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:** df(pandas.DataFrame): screener information table

**getFilterOptions** (*screen\_filter*)

Get filters options.

**Args:** screen\_filter(str): screen filter for checking options

**Returns:** filter\_options(list): all the available filters

**getFilters** ()

Get filters.

**Returns:** filters(list): all the available filters

**getOrders** ()

Get orders.

**Returns:** orders(list): all the available orders

**getSignal** ()

Get signals.

**Returns:** signals(list): all the available trading signals

**set\_filter** (*signal=", filters\_dict={}, ticker="*)

Update the settings.

**Args:** 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** (*verbose=1*)  
Get screener table.

**Args:** verbose(int): choice of visual the progress. 1 for visualize progress.

**Returns:** tickers(list): get all the tickers as 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.

**Args:** group(str): choice of group option order(str): sort the table by the choice of order

**Returns:** df(pandas.DataFrame): group information table

**getGroup** ()  
Get groups.

**Returns:** groups(list): all the available groups

```
getOrders ()  
    Get orders.  
  
    Returns: orders(list): all the available orders
```

### 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.  
  
    ScreenerView (group='Sector', order='Name', out_dir="")  
        Get screener table.  
  
    Args: 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(dict): news table
```

## 6.5 Insider

Getting trading insider.

```
class finvizfinance.insider.Insider (option='latest')  
    Getting information from the finviz insider page.  
  
    Args: option (str): choose a option (latest, top week, top owner trade, insider_id)  
  
    getInsider ()  
        Get insider information table.  
  
    Returns: df(pandas.DataFrame): insider information table
```

## 6.6 Forex

Getting foreign exchange information.

**class** finvizfinance.forex.Forex

Getting information from the finviz forex page.

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

Get forex chart.

**Args:** forex (str): foreign exchange name timeframe (str): choice of timeframe(5M, H, D, W, M)

**performance** (change='percent')

Get forex performance table.

**Args:** change (str): choose an option of change(percent, PIPS)

**Returns:** df(pandas.DataFrame): forex performance table

## 6.7 Crypto

Getting crypto currency information.

**class** finvizfinance.crypto.Crypto

Getting information from the finviz crypto page.

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

Get crypto chart.

**Args:** crypto (str): crypto currency timeframe (str): choice of timeframe(5M, H, D, W, M)

**performance** ()

Get crypto performance table.

**Returns:** df(pandas.DataFrame): crypto performance table

## 6.8 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**

**Args:**

**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.

**Args:** output\_dir(str): name of the output directory.

**outputExcel** (output\_file='earning\_days.xlsx')

Output dataframes to single Excel file.

**Args:** output\_file(str): name of the output excel file.

**partitionDays** (*mode*='financial')

Partition dataframe to separate dataframes according to the dates.

**Args:**

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

## CHAPTER 7

---

### Indices and tables

---

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





### f

- `finvizfinance.crypto`, [17](#)
- `finvizfinance.earnings`, [17](#)
- `finvizfinance.forex`, [17](#)
- `finvizfinance.group.overview`, [15](#)
- `finvizfinance.group.performance`, [16](#)
- `finvizfinance.group.spectrum`, [16](#)
- `finvizfinance.group.valuation`, [16](#)
- `finvizfinance.insider`, [16](#)
- `finvizfinance.news`, [16](#)
- `finvizfinance.quote`, [13](#)
- `finvizfinance.screener.financial`, [15](#)
- `finvizfinance.screener.overview`, [14](#)
- `finvizfinance.screener.ownership`, [15](#)
- `finvizfinance.screener.technical`, [15](#)
- `finvizfinance.screener.ticker`, [15](#)
- `finvizfinance.screener.valuation`, [15](#)



## C

chart () (*finvizfinance.crypto.Crypto* method), 17  
 chart () (*finvizfinance.forex.Forex* method), 17  
 compare () (*finvizfinance.screener.overview.Overview* method), 14  
 Crypto (class in *finvizfinance.crypto*), 17

## E

Earnings (class in *finvizfinance.earnings*), 17

## F

Financial (class in *finvizfinance.screener.financial*), 15  
 finvizfinance (class in *finvizfinance.quote*), 13  
 finvizfinance.crypto (module), 17  
 finvizfinance.earnings (module), 17  
 finvizfinance.forex (module), 17  
 finvizfinance.group.overview (module), 15  
 finvizfinance.group.performance (module), 16  
 finvizfinance.group.spectrum (module), 16  
 finvizfinance.group.valuation (module), 16  
 finvizfinance.insider (module), 16  
 finvizfinance.news (module), 16  
 finvizfinance.quote (module), 13  
 finvizfinance.screener.financial (module), 15  
 finvizfinance.screener.overview (module), 14  
 finvizfinance.screener.ownership (module), 15  
 finvizfinance.screener.technical (module), 15  
 finvizfinance.screener.ticker (module), 15  
 finvizfinance.screener.valuation (module), 15  
 Forex (class in *finvizfinance.forex*), 17

## G

getFilterOptions () (*finvizfinance.screener.overview.Overview* method), 14  
 getFilters () (*finvizfinance.screener.overview.Overview* method), 14  
 getGroup () (*finvizfinance.group.overview.Overview* method), 15  
 getInsider () (*finvizfinance.insider.Insider* method), 16  
 getNews () (*finvizfinance.news.News* method), 16  
 getOrders () (*finvizfinance.group.overview.Overview* method), 15  
 getOrders () (*finvizfinance.screener.overview.Overview* method), 14  
 getSignal () (*finvizfinance.screener.overview.Overview* method), 14

## I

Insider (class in *finvizfinance.insider*), 16

## N

News (class in *finvizfinance.news*), 16

## O

outputCSV () (*finvizfinance.earnings.Earnings* method), 17  
 outputExcel () (*finvizfinance.earnings.Earnings* method), 17  
 Overview (class in *finvizfinance.group.overview*), 15  
 Overview (class in *finvizfinance.screener.overview*), 14  
 Ownership (class in *finvizfinance.screener.ownership*), 15

## P

partitionDays () (*finvizfinance.earnings.Earnings* method), 17

Performance (class in finvizfinance.group.performance), 16  
 performance() (finvizfinance.crypto.Crypto method), 17  
 performance() (finvizfinance.forex.Forex method), 17

## S

ScreenableView() (finvizfinance.group.overview.Overview method), 15  
 ScreenableView() (finvizfinance.group.spectrum.Spectrum method), 16  
 ScreenableView() (finvizfinance.screener.overview.Overview method), 14  
 ScreenableView() (finvizfinance.screener.ticker.Ticker method), 15  
 set\_filter() (finvizfinance.screener.overview.Overview method), 14  
 Spectrum (class in finvizfinance.group.spectrum), 16

## T

Technical (class in finvizfinance.screener.technical), 15  
 Ticker (class in finvizfinance.screener.ticker), 15  
 TickerCharts() (finvizfinance.quote.finvizfinance method), 13  
 TickerDescription() (finvizfinance.quote.finvizfinance method), 13  
 TickerFullInfo() (finvizfinance.quote.finvizfinance method), 13  
 TickerFundament() (finvizfinance.quote.finvizfinance method), 13  
 TickerInsideTrader() (finvizfinance.quote.finvizfinance method), 13  
 TickerNews() (finvizfinance.quote.finvizfinance method), 13  
 TickerOuterRatings() (finvizfinance.quote.finvizfinance method), 14

## V

Valuation (class in finvizfinance.group.valuation), 16  
 Valuation (class in finvizfinance.screener.valuation), 15