

Shape your world



Ritsumeikan
Asia Pacific University

OPENBB TERMINAL

MANUAL FOR DUMMIES

- Data Mining, Machine Learning, NLP -



“Your key to a free, but comprehensive finance platform”

Authors:

12419061 - PHAM Le Cong Kien - ph19l6jp@apu.ac.jp

12419205 – POKHAREL Sumit - po19s6pd@apu.ac.jp

12419032 – HUYNH Vu Thao Nguyen - hu19v6im@apu.ac.jp

Supervisor:

CLASTER William Bill – wclaster@apu.ac.jp

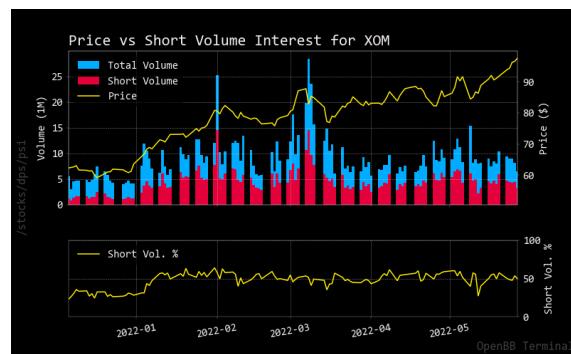
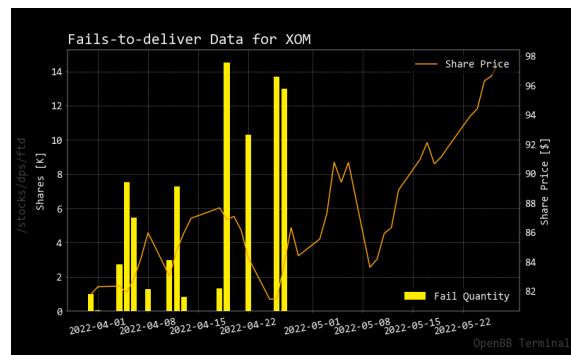
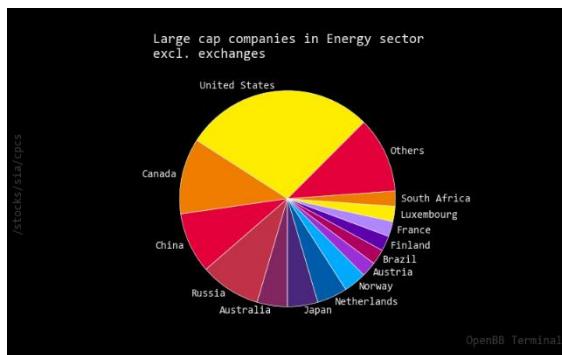
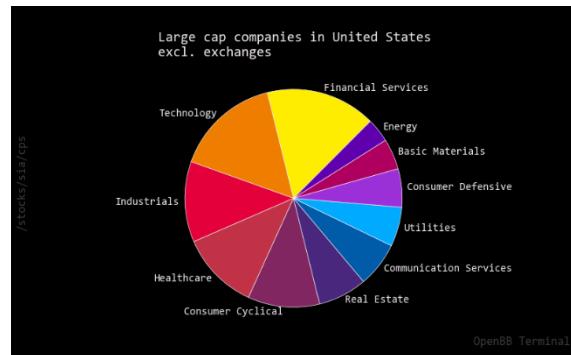
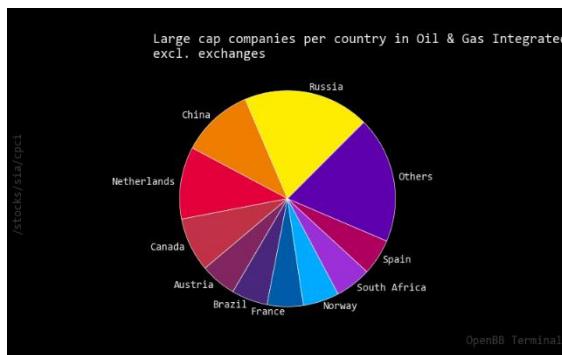
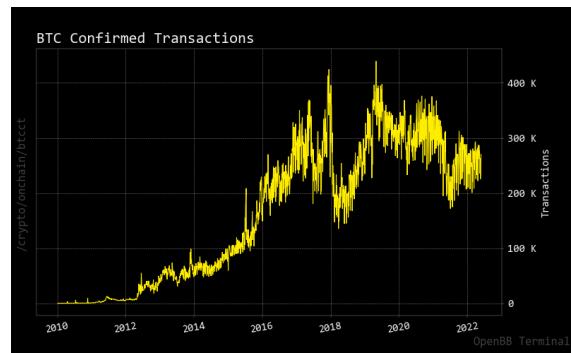
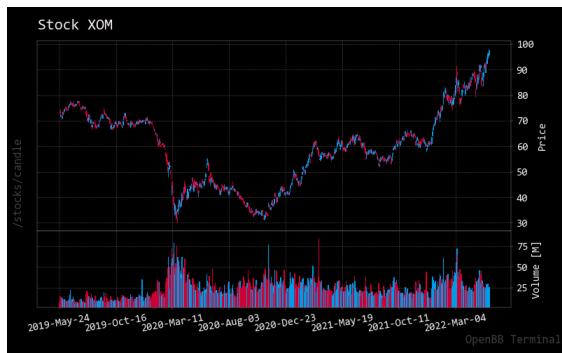


Table of Contents

1 Overview	3
1.1 Introduction	3
1.2 Setup	4
1.3 OpenBB Terminal 101	5
2 Stocks	7
2.1 General Information	7
2.2 Digging Deeper	12
3 Cryptocurrency	17
3.1 General Information	17
3.2 Digging Deeper	18
4 Forex	21
4.1 General Information	21
4.2 Digging Deeper	24
5 Econometrics	26
5.1 General Information	26
5.2 Digging Deeper	28
Appendix	32
API Keys	37



OpenBB Terminal Preview





1 OVERVIEW

1.1 Introduction

In the current world of finance, in order to monitor and analyze real-time financial market data such as news, price quotes, and messaging across its proprietary secure network, Bloomberg Terminal is ready to equip investors and professionals with the information. However, the cost for Bloomberg Terminal equals to around \$2400 per month, which is a high price for the amateurs. With that in mind, **OpenBB Terminal** is created to give investment chances for everyone, regardless of who they are and where they are.

OpenBB Terminal, formerly known as *Gamestonk Terminal*, is a rising Python-based open-source investment analysis platform. Dedicated to financing and investing experts, **OpenBB Terminal** provides the environment for investment research which allows an average joe retail trader to leverage state-of-the-art Economic and financial analysis, as well as analytics, data science, and machine learning technologies -all in one screen and from one's own personal computer.

As being a platform built on Python, **OpenBB Terminal** provides access to several data libraries in Data Science, Machine Learning, and Data Acquisition as a modern Python-based environment.

Towards financing and investing experts, **OpenBB Terminal** may be a Swiss Army Knife for finance as it provides the following functionalities.

- **Discover Stocks:** Some features are: Top gainers; Sectors performance such as FRED data and balance sheets, income statements, and economic data; upcoming earnings releases; top high shorted interest stocks; top stocks with low float; top orders on fidelity; and some SPAC websites with news/calendars.
- **Market Sentiment:** The main feature is extracting trending symbols on *Stocktwits*, or even *Stocktwits* sentiment based on bull/bear flags; Reddit main posts and most ticker mentions; AI-based sentiment prediction on Twitter; Over Time, Google makes mentions.
- **Research Web Pages:** The terminal provides the lists of good pages to do research on stock.



- **Fundamental Analysis:** The function provides users to read financials of a company from *Market Watch*, *Yahoo Finance*, *Alpha Vantage*, and *Financial Modelling Prep API*.
- **Technical Analysis:** The usual technical indicators: `sma`, `rsi`, `macd`, `adx`, `bbands`, and more.
- **Due Diligence:** There are useful features equipped: Latest news of the company; Analyst prices and ratings; Price target from several analysts plot over time vs stock price; Insider activity, and these timestamps marked on the stock price historical data; Latest SEC filings; Short interest over time; A check for financial warnings based on Sean Seah book.
- **Prediction Techniques:** The Terminal provides prediction on the stock price, from simple models like `sma` and `arima` to complex neural network models, like *LSTM*.
- **Reports:** This feature allows user to run several jobs functionalities and write daily notes on a stock; therefore, user can better their analysis about the stock in the past, to perform better decisions.
- **Comparison Analysis:** OpenBB Terminal provides features allowing users to compare stocks in the same industries.
- **On the ROADMAP:** The Terminal not only provides the information about the stocks, but also provides that of *Cryptocurrencies*, *Portfolio Analysis*, *Credit Analysis*, etc.

As being an open-source platform, **OpenBB Terminal** may continue to grow as the time being to allow more people to take advantage of.

With that in mind, we hope that this **OpenBB Terminal**'s manual will provide our readers an overview of how to make use of the terminal for better analysis and decision in terms of finance and investment.

1.2 Setup

Setup of **OpenBB Terminal** is simple for the typical user. Simply download the program and install it. Typically, the setup takes less than 5 minutes. Alternatives for programmers and contributors to the OpenBB project include:

- Using *Installer*: recommended if users just want to use the Terminal only.
- Using *Python*: recommended if users want to develop a new feature.



- Using *Docker*: alternative option to the installer if preferred.
- Using *Docker Web UI*: if users want to deploy the web UI to access over their LAN.

However, in this manual, we will only describe the first option: using installer, which is the simplest way to make use of the platform for economic and financial analysis and analytics.

[The details will be provided in the appendix]

1.3 OpenBB Terminal 101

OpenBB Terminal may look a bit unnerving to the first-time user, however the pattern of accessing data is simple and uniform. Upon opening OpenBB, we see a list of actions that we can perform. Although they are listed in like a command-line terminal, one can just think of them as a menu we may access a typical program. For example, in Excel we have a File menu where we can access certain capabilities like creating a new document or opening an existing document etc.

The first "menu" we see looks like this:

```
The main commands you should be aware when navigating through the terminal are:
cls           clear the screen
help / h / ?   help menu
quit / q / ..  quit this menu and go one menu above
exit          exit the terminal
reset / r      reset the terminal and reload configs from the current location
resources      only available on main contexts (not sub-menus)

about          about us
update         update terminal automatically
tz             set different timezone
export         select export folder to output data
exe            execute automated routine script
> settings    set feature flags and style charts
> keys        set API keys and check their validity
```

This looks quite cryptic and straight out of the 90's at first glance; people give up right away when they see something like this. We ourselves are no experts, and it took us some time to get our eyes used to this as well. However, we have spent all this time to make this manual just so that it becomes easy for you to navigate. **We will go through each step one-by-one and provide you with clear helping hand each time so that you always stay in track.**

From the above figure, we see that we have access to a rich source of data and information. OpenBB Terminal is a wealth of information on world economies, econometrics,



and finance. Specifically, we can choose now to explore **stocks**, **crypto**, **etfs**, **economy**, **forex**, **funds**, **alternative data**, **econometrics**, **portfolio data**. Once we select one of these items, we will see a new menu of options related to that which we have chosen to explore. Again, a little cryptic but definitely very comprehensible.

```
Export Folder: DEFAULT (folder: exports/)  
Timezone: US/Eastern  
  
> stocks  
> crypto  
> etf  
> economy  
> forex  
> funds  
> alternative  
> econometrics  
> portfolio  
> jupyter
```

OpenBB Terminal —

For instance, if we want to discover and analyse how the stock market has been going, we simply need to type **stocks** after the automatically registered \$ sign. A new menu of various options will appear as follows. This, too, may look a bit intimidating at first (believe us, we get your worry) but if we take a closer look, we can see that there are just commands that we can utilize.

```
2022 May 28, 15:06 (✉ ) / $ stocks  
Stocks —  
  
search      search a specific stock ticker for analysis  
load        load a specific stock ticker and additional info for analysis  
  
Stock: Daily  
  
Datetime:  
Timezone:  
Exchange:  
Market:  
Currency:  
  
quote       view the current price for a specific stock ticker  
candle      view a candle chart for a specific stock ticker  
news        latest news of the company [News API]  
  
> options    options menu,          e.g.: chains, open interest, greeks, parity  
> disc       discover trending stocks,   e.g.: map, sectors, high short interest  
> sia        sector and industry analysis, e.g.: companies per sector, quick ratio per industry and country  
> dps        dark pool and short data,   e.g.: darkpool, short interest, ftd  
> scr        screener stocks,       e.g.: overview/performance, using preset filters  
> ins        insider trading,        e.g.: latest penny stock buys, top officer purchases  
> gov        government menu,       e.g.: house trading, contracts, corporate lobbying  
> ba         behavioural analysis,  from: reddit, stocktwits, twitter, google  
> ca         comparison analysis,  e.g.: get similar, historical, correlation, financials  
> fa         fundamental analysis, e.g.: income, balance, cash, earnings  
> res        research web page,    e.g.: macroaxis, yahoo finance, fool  
> dd         in-depth due-diligence, e.g.: news, analyst, shorts, insider, sec  
> bt         strategy backtester,  e.g.: simple ema, ema cross, rsi strategies  
> ta         technical analysis,  e.g.: ema, macd, rsi, adx, bbands, obv  
> qa         quantitative analysis, e.g.: decompose, cusum, residuals analysis  
> pred       prediction techniques, e.g.: regression, arima, rnn, lstm
```



Or if we want to search for what kind of cryptocurrency is trending right now for example, we simply type `crypto` after the `$` sign, similar to the aforementioned stocks function. A menu of various options related to cryptocurrency will appear as follows (just normal commands again by the way!). Regarding how to delve further into the menu itself, we will explain in detail later.

```
2022 May 15, 03:57 (✉ ) / $ crypto
                               Cryptocurrency
load      load a specific cryptocurrency for analysis
find     find coins

Coin:
Source:

headlines  crypto sentiment from 15+ major news headlines [Finbrain]
chart      view a candle chart for a specific cryptocurrency
prt        potential returns tool - check how much upside if ETH reaches BTC market cap

> disc      discover trending cryptocurrencies,          e.g.: top gainers, losers, top sentiment
> ov        overview of the cryptocurrencies,          e.g.: market cap, DeFi, latest news, top exchanges, stab
> onchain   information on different blockchains,    e.g.: eth gas fees, whale alerts, DEXes info
> defi      decentralized finance information,       e.g.: dpi, llama, tvl, lending, borrow, funding
> tools     explore different tools                 e.g.: apytoapr, il
> nft       non-fungible tokens,                     e.g.: today drops
> dd        due-diligence for loaded coin,          e.g.: coin information, social media, market stats
> ta        technical analysis for loaded coin,      e.g.: ema, macd, rsi, adx, bbands, obv
> pred     prediction techniques,                  e.g.: regression, arima, rnn, lstm, conv1d, monte carlo
> qa        quantitative analysis,                 e.g.: decompose, cusum, residuals analysis

                               OpenBB Terminal v1.3.0m (https://openbb.co)
2022 May 15, 04:06 (✉ ) /crypto/ $ -
```

To go up/back to the previous menu, we can just type `quit` (or `q`) and we will return to the original menu where we saw stocks, crypto, etf, etc.

This is the overview of how we will navigate around OpenBB Terminal. Now, let us begin with the heart of this terminal, i.e., the details within each function.



2 STOCKS

2.1 General Information

In order to access the Stocks data, we can simply type `stocks` on the terminal. Please make sure to go back to the home (simply type `home` command on the left-hand-side) of the terminal and execute this. The following window will appear (a bit cryptic again but these commands are very interesting).

```
[2022 May 15, 07:40 (🦋) / $ stocks
Stocks

search      search a specific stock ticker for analysis
load        load a specific stock ticker and additional info for analysis

Stock: Daily

Datetime:
Timezone:
Exchange:
Market:
Currency:

quote       view the current price for a specific stock ticker
candle      view a candle chart for a specific stock ticker
news        latest news of the company [News API]

> options    options menu,                                     e.g.: chains, open interest, greeks, parity
> disc       discover trending stocks,                   e.g.: map, sectors, high short interest
> sia        sector and industry analysis,             e.g.: companies per sector, quick ratio per industry and
> dps        dark pool and short data,                 e.g.: darkpool, short interest, ftd
> scr        screener stocks,                         e.g.: overview/performance, using preset filters
> ins        insider trading,                        e.g.: latest penny stock buys, top officer purchases
> gov        government menu,                      e.g.: house trading, contracts, corporate lobbying
> ba         behavioural analysis,                from: reddit, stocktwits, twitter, google
> ca         comparison analysis,                  e.g.: get similar, historical, correlation, financials
> fa         fundamental analysis,            e.g.: income, balance, cash, earnings
> res        research web page,                  e.g.: macroaxis, yahoo finance, fool
> dd         in-depth due-diligence,           e.g.: news, analyst, shorts, insider, sec
> bt         strategy backtester,            e.g.: simple ema, ema cross, rsi strategies
> ta         technical analysis,              e.g.: ema, macd, rsi, adx, bbands, obv
> qa         quantitative analysis,          e.g.: decompose, cusum, residuals analysis
> pred      prediction techniques,          e.g.: regression, arima, rnn, lstm

2022 May 15, 07:41 (🦋) /stocks/ $ ]
```

We can get the general information of a stock via the help of the commands such as `load`, `quote`, `candle`, and `news`. Point to note that if we want to get information about a stock, we need to **load it first** (picture below; pay attention to `$ load TSLA`). When we load a stock, the terminal will continue to remember that stock until we either exit the terminal or change the stock by manually loading a different stock as long as the stock command is in function.



Stocks

```
search      search a specific stock ticker for analysis
load       load a specific stock ticker and additional info for analysis

Stock: Daily

Datetime:
Timezone:
Exchange:
Market:
Currency:

quote      view the current price for a specific stock ticker
candle     view a candle chart for a specific stock ticker
news       latest news of the company [News API]

> options   options menu,                               e.g.: chains, open interest, greeks, parity
> disc      discover trending stocks,               e.g.: map, sectors, high short interest
> sia       sector and industry analysis,          e.g.: companies per sector, quick ratio per industry and
> dps       dark pool and short data,              e.g.: darkpool, short interest, ftd
> scr       screener stocks,                      e.g.: overview/performance, using preset filters
> ins       insider trading,                     e.g.: latest penny stock buys, top officer purchases
> gov       government menu,                   e.g.: house trading, contracts, corporate lobbying
> ba        behavioural analysis,            from: reddit, stocktwits, twitter, google
> ca        comparison analysis,           e.g.: get similar, historical, correlation, financials
> fa        fundamental analysis,          e.g.: income, balance, cash, earnings
> res       research web page,             e.g.: macroaxis, yahoo finance, fool
> dd        in-depth due-diligence,        e.g.: news, analyst, shorts, insider, sec
> bt        strategy backtester,         e.g.: simple ema, ema cross, rsi strategies
> ta        technical analysis,          e.g.: ema, macd, rsi, adx, bbands, obv
> qa        quantitative analysis,    e.g.: decompose, cusum, residuals analysis
> pred      prediction techniques,   e.g.: regression, arima, rnn, lstm
```

```
[2022 May 15, 14:31 (🦋) /stocks/ $ load TSLA
Loading Daily TSLA stock with starting period 2019-05-13 for analysis.

Datetime: 2022 May 15 14:31
Timezone: America/New_York
Currency: USD
Market: CLOSED
```

Quoting a stock: We either enter the stock symbol **AFTER** we have loaded the stock (above figure) OR we simply type **quote STOCK_SYMBOL** on the left-hand-side (figure below). **Again, we are going through each process step-by-step so please do not worry about these tangled pictures.**



```
Stocks

search      search a specific stock ticker for analysis
load        load a specific stock ticker and additional info for analysis

Stock: Daily

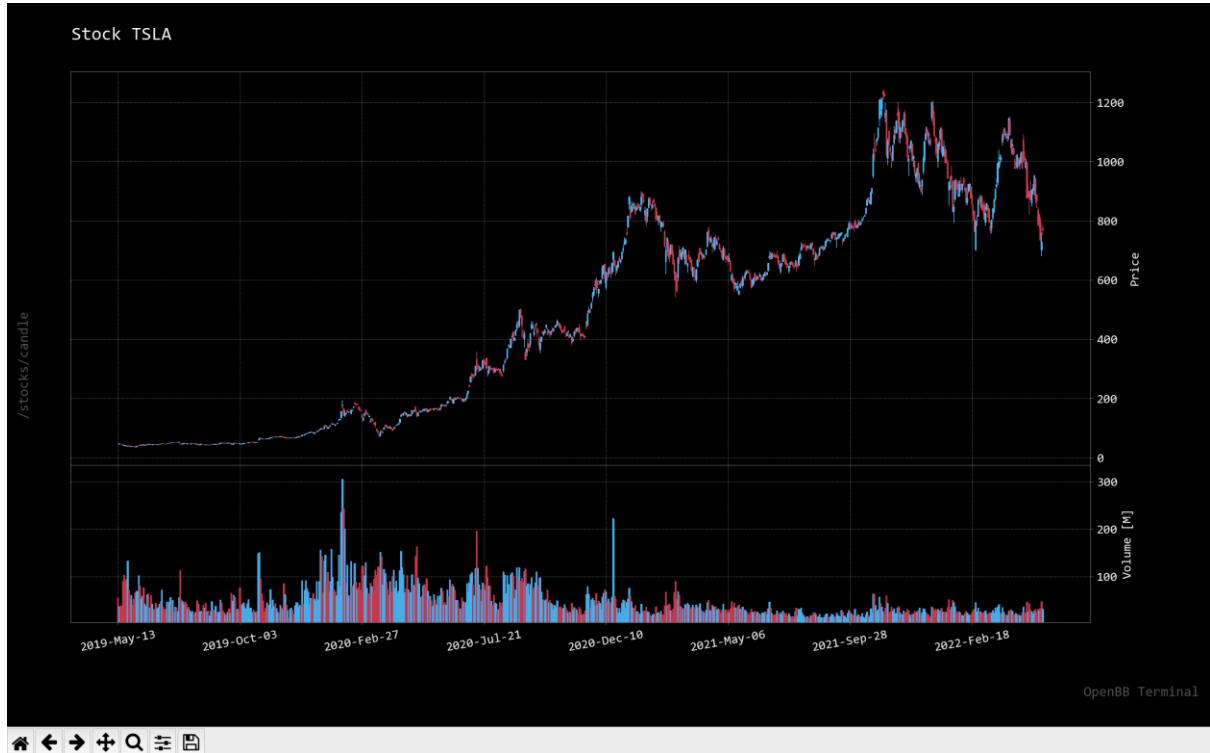
Datetime:
Timezone:
Exchange:
Market:
Currency:

quote       view the current price for a specific stock ticker
candle      view a candle chart for a specific stock ticker
news        latest news of the company [News API]

> options    options menu,                                e.g.: chains, open interest, greeks, parity
> disc       discover trending stocks,                 e.g.: map, sectors, high short interest
> sia        sector and industry analysis,            e.g.: companies per sector, quick ratio per industry and country
> dps        dark pool and short data,                e.g.: darkpool, short interest, ftd
> scr        screener stocks,                         e.g.: overview/performance, using preset filters
> ins        insider trading,                         e.g.: latest penny stock buys, top officer purchases
> gov        government menu,                         e.g.: house trading, contracts, corporate lobbying
> ba         behavioural analysis,                  from: reddit, stocktwits, twitter, google
> ca         comparison analysis,                  e.g.: get similar, historical, correlation, financials
> fa         fundamental analysis,                 e.g.: income, balance, cash, earnings
> res        research web page,                     e.g.: macroaxis, yahoo finance, fool
> dd         in-depth due-diligence,                e.g.: news, analyst, shorts, insider, sec
> bt         strategy backtester,                  e.g.: simple ema, ema cross, rsi strategies
> ta         technical analysis,                   e.g.: ema, macd, rsi, adx, bbands, obv
> qa         quantitative analysis,              e.g.: decompose, cusum, residuals analysis
> pred       prediction techniques,             e.g.: regression, arima, rnn, lstm

2022 May 15, 14:43 (🦋) /stocks/ $ quote TSLA
Symbol      TSLA
Name        Tesla, Inc.
Price       769.59
Open        773.48
High        787.35
Low         751.57
Previous Close 728.00
Volume     30,716,988
52 Week High 1243.49
52 Week Low  546.98
Change      41.59
Change %    5.71%
```

We can similarly load a stock candle **AFTER** we have loaded the stock by typing `candle` on the left-hand-side. A candle chart will pop up on our screen (figure below). We can magnify this chart and save it as well.



We can also look up some news insights on a publicly traded company with the help of the news command. We can load the news **AFTER** we have loaded the stock by typing **news** on the left-hand-side (figure below). Important to note that we cannot load the news unless the news API key has been defined by the terminal (refer to the **API Keys** in the Appendix section for details).

```
2022 May 15, 14:50 (TF) /stocks/ $ candle
2022 May 15, 14:50 (TF) /stocks/ $ news
481 news articles for Tesla,+Inc. were found since 2022-05-09
2022-05-15 17:35:24 CEO pay packages rose to $14.7 million in 2021, a new high
https://www.livemint.com/companies/news/ceo-pay-packages-rose-to-14-7-million-in-2021-a-new-high-11652620345989.html
2022-05-15 17:24:53 Covid-Hit Shanghai To Allow Gradual Reopening Of Malls And Markets, Xinhua Says
https://www.forbes.com/sites/russellflannery/2022/05/15/covid-hit-shanghai-to-allow-gradual-reopening-of-malls-and-markets-xinhua-says/
2022-05-15 14:45:00 ROSEN, A LEADING AND TOP RANKED LAW FIRM, Encourages Twitter, Inc. Investors with Losses Over $100K to Secure Counsel Before Important Deadline in Securities Class Action Against Elon Musk – TWTR
https://www.prnewswire.com/news-releases/rosen-a-leading-and-top-ranked-law-firm-encourages-twitter-inc-investors-with-losses-over-100k-to-secure-counsel-before-important-deadline-in-securities-class-action-against-elon-musk--twtr-301547361.html
2022-05-15 13:07:38 Legal team told him he violated an NDA...
https://news.yahoo.com/elon-musk-says-twitter-legal-000100024.html
2022-05-15 12:16:00 China Adds Support to Property Market as Shanghai Reopens Shops
https://www.bloomberg.com/news/articles/2022-05-15/shanghai-adds-1-369-local-covid-cases-zero-outside-quarantine
```

We can go deeper into the news by undergoing specific commands, such as fixing a particular starting date for the news, listing the oldest articles first, article sources, and so on. Type **news -h** for details (figure below).



```
2022 May 15, 14:55 (🦋) /stocks/ $ news -h
usage: news [-l LIMIT] [-d N_START_DATE] [-o] [-s SOURCES [SOURCES ...]] [-h]

Prints latest news about company, including date, title and web link. [Source: News API]

optional arguments:
-l LIMIT, --limit LIMIT
    Limit of latest news being printed.
-d N_START_DATE, --date N_START_DATE
    The starting date (format YYYY-MM-DD) to search articles from
-o, --oldest
    Show oldest articles first
-s SOURCES [SOURCES ...], --sources SOURCES [SOURCES ...]
    Show news only from the sources specified (e.g bbc yahoo.com)
-h, --help
    show this help message
```

The flag `-h` can simply be typed after any kind of command and the terminal will provide us with different kinds of command arguments (in the above picture, we have `-l LIMIT`, `-d N_START_DATE`, `-o`, `-s`, and `-h` itself) we can employ within the command itself. This is a very helpful tool for people who are trying to get familiar with OpenBB Terminal. Each command has its own set of arguments so we are bound to find new things as we keep exploring the terminal.

Additionally, the command `-h` is a help menu command that can be typed on the left-hand-side after the `$` symbol (at any instant) to get the list of commands and tools we can utilize to extract information.

2.2 Digging Deeper

There are countless things we can do within the `stocks` entry that can be seen when we load it. The way of digging deeper into the `stocks` entry (or any entry for that matter) is by simply getting used to loading the commands and by using the help command – either `-h` or `h` – whenever necessary; we will always keep discovering new functions. Let's go through some interesting ones.

disc (discover trending stocks)

Let's load `disc` within `stocks`. The following window will appear.



```
[2022 May 15, 15:20 (W) /stocks/ $ disc
                                                Stocks - Discovery ----

[Finnhub]
  pipo      past IPOs dates
  fipo      future IPOs dates
[Yahoo Finance]
  gainers   show latest top gainers
  losers    show latest top losers
  ugs       undervalued stocks with revenue and earnings growth in excess of 25%
  gtech    tech stocks with revenue and earnings growth more than 25%
  active   most active stocks by intraday trade volume
  ulc      potentially undervalued large cap stocks
  asc      small cap stocks with earnings growth rates better than 25%
[Fidelity]
  ford     orders by Fidelity Customers
[Cathiesark.com]
  arkord   orders by ARK Investment Management LLC
[Seeking Alpha]
  upcoming  upcoming earnings release dates
  trending  trending news
  cnews    customized news (buybacks, ipos, spacs, healthcare, politics)
[Shortinterest.com]
  lowfloat low float stocks under 10M shares float
[Pennystockflow.com]
  hotpenny today's hot penny stocks
[NASDAQ Data Link (Formerly Quandl)]
  rtat     top 10 retail traded stocks per day
  divcal   dividend calendar for selected date
```

We can see a lot of commands that we can utilize to extract different kinds of information. Note that it is possible to load `-pipo` and `-fipo` only after the Finnhub API key has been defined by the terminal (refer to the [API Keys](#) in the Appendix section for details). Let's try having a look at `gainers`.

Load `gainers` within the `disc` command. Something similar to the following window will appear.

Symbol	Name	Price (Intraday)	Change	% Change	Volume	Avg Vol (3 month)	Market Cap	PE Ratio (TTM)
0 DUOL	Duolingo, Inc.	89.77	22.79	+34.03%	2.016M	533940	3.518	
1 LSPD	Lightspeed Commerce Inc.	21.37	5.20	+32.16%	2.865M	1.242M	3.172B	
2 FLMMF	Filo Mining Corp.	16.69	4.04	+31.94%	82860	27657	2.027B	
3 PAGS	PagSeguro Digital Ltd.	13.47	3.25	+31.88%	9.683M	4.367M	4.417B	19.55
4 AFRM	Affirm Holdings, Inc.	23.71	5.67	+31.43%	84.939M	13.263M	6.776B	

On May 15, 2022, Duolingo, Inc. ranked number one in the latest top gainers' section. Though the terminal displays only 4 companies, we can get the information of up to 25 companies if we export this data (we can apply the `-l` flag to filter the data to our desires as well; see figure below).

To export, we can simply refer to the `-h` flag after the `gainers` command (see figure below).

```
[2022 May 15, 15:21 (W) /stocks/disc/ $ gainers -h
usage: gainers [-l LIMIT] [-h] [--export EXPORT]

Print up to 25 top gainers. [Source: Yahoo Finance]

optional arguments:
  -l LIMIT, --limit LIMIT
                        Limit of stocks to display. (default: 5)
  -h, --help           show this help message (default: False)
  --export EXPORT      Export raw data into csv, json, xlsx (default: )
```



--export is the flag used to export information from the terminal to our computer. We can verify whether we can use this flag on a specific command or not with the help of the -h flag. Note that the choices for the data format via the --export flag will vary according to the data we are going to export. For the gainers command, we are able to export the data into .csv, .json, and .xlsx format.

Let's export the gainers data into a csv file in our computer (figure below).

```
2022 May 15, 15:21 (🦋) /stocks/disc/ $ gainers --export csv
Symbol          Name   Price (Intraday) Change % Change  Volume Avg Vol (3 month) Market Cap PE Ratio (TTM)
0 DUOL          Duolingo, Inc.    89.77  22.79  +34.03%  2.016M      533940  3.51B
1 LSPD          Lightspeed Commerce Inc. 21.37   5.20  +32.16%  2.865M      1.242M  3.172B
2 FLMMF         Filo Mining Corp.  16.69   4.04  +31.94%  82860       27657   2.027B
3 PAGS          PagSeguro Digital Ltd. 13.47   3.25  +31.80%  9.603M      4.367M  4.417B
4 AFRM          Affirm Holdings, Inc. 23.71   5.67  +31.43%  84.939M     13.263M  6.776B
Saved file: /Users/psumit/Desktop/OPENBB-exports/20220516_042143_stocks_discovery_gainers.csv
```

Note that the terminal tells us where exactly the file has been exported (in the figure above, pay attention to [Saved file: /Users/...](#))

Exported file's screenshot below:

	A	B	C	D	E	F	G	H	I	J	K
1	Symbol	Name	Price (Intraday)	Change	% Change	Volume	Avg Vol (3 month)	Market Cap	PE Ratio (TTM)		
2	0 DUOL	Duolingo, Inc.	89.77	22.79	34.03%	2.016M	533940	3.51B			
3	1 LSPD	Lightspeed Commerce Inc.	21.37	5.20	+32.16%	2.865M	1.242M	3.172B			
4	2 FLMMF	Filo Mining Corp.	16.69	4.04	+31.94%	82860	27657	2.027B			
5	3 PAGS	PagSeguro Digital Ltd.	13.47	3.25	+31.80%	9.603M	4.367M	4.417B	19.55		
6	4 AFRM	Affirm Holdings, Inc.	23.71	5.67	+31.43%	84.939M	13.263M	6.776B			
7	5 HOOD	Robinhood Markets, Inc.	10.69	2.13	+24.88%	48.56M	21.084M	8.933B			
8	6 EQRX	EQRx, Inc.	4.11	0.8	+24.17%	3.676M	1.409M	2.004B			
9	7 ROIV	Rovant Sciences Ltd.	3.46	0.65	+23.13%	1.096M	438347	2.395B			
10	8 ZLAB	Zai Lab Limited	33.22	6.15	+22.72%	901660	871403	3.253B			
11	9 SE	Sea Limited	75.4	13.95	+22.70%	17.415M	8.953M	42.204B			
12	10 ASAN	Asana, Inc.	24.37	4.5	+22.65%	7.387M	5.342M	4.633B			
13	11 NUTX	Nutex Health, Inc.	6.12	1.12	+22.40%	3.93M	5.849M	3.94B	1.87		
14	12 COMP	Compass, Inc.	5.45	0.98	+21.92%	17.016M	3.316M	2.327B			
15	13 ONTTF	Oxford Nanopore Technologies Plc	3.5799	0.6299	+21.35%	17823	15608	2.946B			
16	14 OTLY	Oatly Group AB	3.71	0.64	+20.85%	9.219M	5.053M	2.195B			
17	15 DLO	DLocal Limited	20.77	3.54	+20.55%	2.988M	1.401M	6.128B	83.08		
18	16 NU	Nu Holdings Ltd.	4.82	0.82	+20.50%	25.636M	13.593M	22.22B			
19	17 JOBY	Joby Aviation, Inc.	5.33	0.9	+20.32%	10.359M	6.779M	3.23B			
20	18 STNE	StoneCo Ltd.	8.72	1.46	+20.11%	10.48M	9.57M	2.688B			
21	19 WE	WeWork Inc.	6.76	1.13	+20.07%	13.255M	4.664M	5.359B			
22	20 MSTR	MicroStrategy Inc.	204.57	33.39	+19.51%	2.523M	632114	2.311B			
23	21 SOFI	SoFi Technologies, Inc.	6.75	1.09	+19.26%	74.842M	57.58M	5.398B			
24	22 SI	Silvergate Capital Corp.	75.87	12.17	+19.11%	3.217M	977886	2.4B	24.27		
25	23 BILL	Bill.com Holdings, Inc.	118.54	18.55	+18.55%	4.91M	2.249M	12.36B			
26	24 OSH	Oak Street Health, Inc.	18	2.79	+18.34%	2.958M	2.679M	4.339B			
27											



We can follow the exact same steps to get the data on **losers** as well.

Let's try **ugs**. Load **ugs** within the **disc** command. Something similar to the following window will appear.

Symbol	Name	Price (Intraday)	Change	% Change	Volume	Avg Vol (3 month)	Market Cap	PE Ratio (TTM)
0 F	Ford Motor Company	13.60	1.06	+8.52%	78.944M	73.364M	54.267B	4.70
1 BAC	Bank of America Corporation	35.17	0.10	+0.29%	49.118M	54.345M	283.368	10.01
2 PFE	Pfizer Inc.	49.92	-0.47	-0.93%	22.424M	28.408M	281.937B	12.95
3 ITUB	Itaú Unibanco Holding S.A.	4.85	0.12	+2.54%	49.865M	48.992M	47.432B	9.15
4 FCX	Freeport-McMoRan Inc.	35.04	0.70	+2.04%	18.988M	18.471M	50.808B	10.14

We can use the **-h** flag to see the functions we can undergo for **ugs** as well. We can export this data in the exact manner we exported the raw data from the **gainers** command.

pred (prediction techniques)

Let's load **pred** within **stocks**. The following window will appear.

```
2022 May 15, 15:54 (W) /stocks/ $ pred
                                         Stocks - Prediction Techniques

load      load new ticker
pick     pick new target variable

Stock: Daily TSLA (from 2019-05-12)
Target Column: AdjClose

Models:
  ets      exponential smoothing (e.g. Holt-Winters)
  knn      k-Nearest Neighbors
  regression polynomial regression
  arima    autoregressive integrated moving average
  mlp      Multilayer Perceptron
  rnn      Recurrent Neural Network
  lstm    Long-Short Term Memory
  conv1d  1D Convolutional Neural Network
  mc      Monte-Carlo simulations

2022 May 15, 15:54 (W) /stocks/pred/ $ █
```

Note that our preloaded stock *TSLA* is still recognized by the terminal. If we want to change the stock, we need to load a new stock (refer to loading the stock within the [2.1 General information](#) sub-section of **Stocks**).

There are various models that we can utilize to predict the stock values of a company. Let's try **regression**. Regression in this case simply refers to the linear regression technique, arguably the most commonly used predictive technique in Statistics, Data Science, Machine Learning, Econometrics, and various other domains. We are not going to explain linear regression here (though it is pretty important and interesting if you want to read it online; highly recommend doing it!). Let's load **regression -h** within **pred**. The following window will appear.



```
[2022 May 15, 16:08 (W) /stocks/pred/ $ regression -h
usage: regression [-i N_INPUTS] [-d N_DAYS] [-j N_JUMPS] [-e S_END_DATE] [-p N_POLYNOMIAL] [-h] [--export EXPORT]

Regression attempts to model the relationship between two variables by fitting a linear/quadratic/cubic/other equation to observed data. One variable is considered to be an explanatory variable, and the other is considered to be a dependent variable.

optional arguments:
-i N_INPUTS, --input N_INPUTS
                number of days to use for prediction.
-d N_DAYS, --days N_DAYS
                prediction days.
-j N_JUMPS, --jumps N_JUMPS
                number of jumps in training data.
-e S_END_DATE, --end S_END_DATE
                The end date (format YYYY-MM-DD) to select - Backtesting
-p N_POLYNOMIAL, --polynomial N_POLYNOMIAL
                polynomial associated with regression.
-h, --help
                show this help message
--export EXPORT
                Export figure into png, jpg, pdf, svg
```

Based on the window, if we want to use the following 5 days for prediction using linear regression, we can simply add `-i 5` after `regression` (figure below).

```
[2022 May 15, 16:12 (W) /stocks/pred/ $ regression -i 5

                                pred
2022-05-16  858.65 $
2022-05-17  975.24 $
2022-05-18  890.13 $
2022-05-19  962.59 $
2022-05-20  998.17 $
```

Note that the prediction numbers change according to the number of days we use for prediction (figure below). A prediction chart will also pop up independently every time we load a prediction.

```
[2022 May 15, 16:14 (W) /stocks/pred/ $ regression -i 5

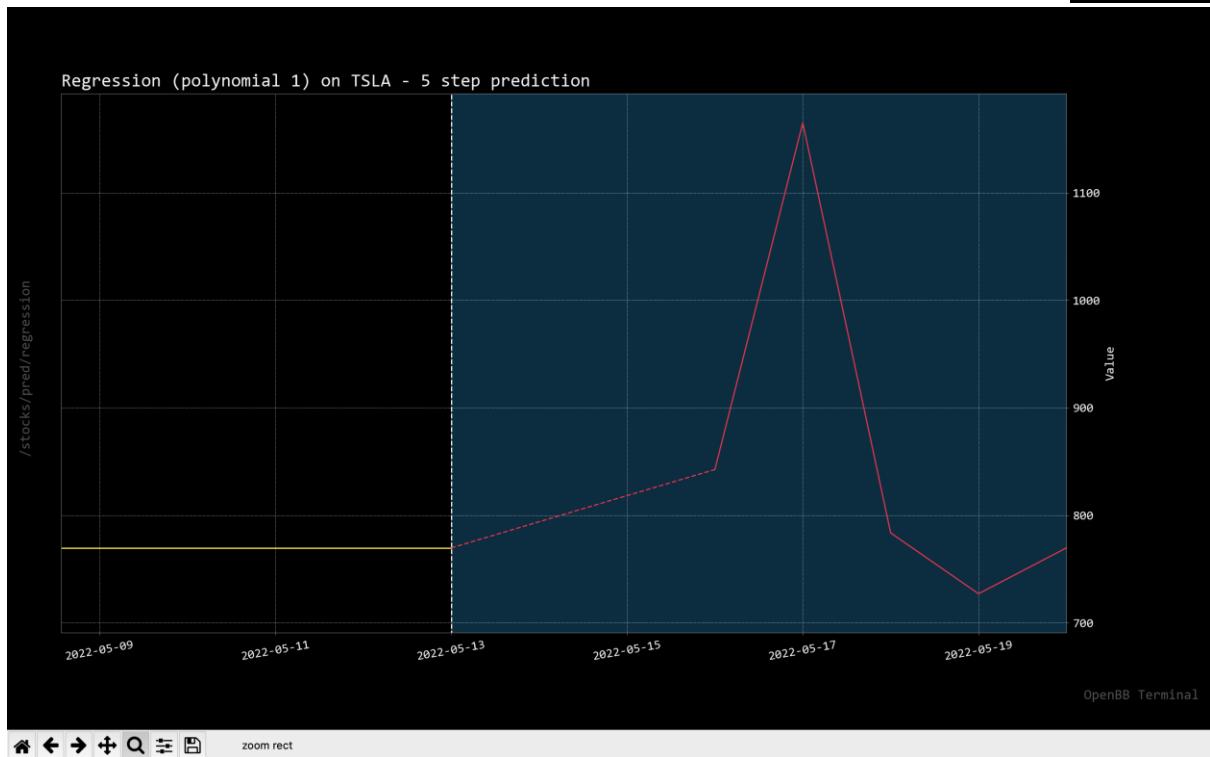
                                pred
2022-05-16  858.65 $
2022-05-17  975.24 $
2022-05-18  890.13 $
2022-05-19  962.59 $
2022-05-20  998.17 $

[2022 May 15, 16:14 (W) /stocks/pred/ $ regression -i 30

                                pred
2022-05-16  842.48 $
2022-05-17  1164.94 $
2022-05-18  783.56 $
2022-05-19  726.94 $
2022-05-20  769.59 $

[2022 May 15, 16:14 (W) /stocks/pred/ $ regression -i 15

                                pred
2022-05-16      566.76 $
2022-05-17  1109.34 $
2022-05-18  1264.11 $
2022-05-19  1285.63 $
2022-05-20  1084.28 $
```



The magnified of the above prediction chart of the polynomial regression was from entering 30 days for the prediction (May 15, 2022 on TSLA).

We can try similar arguments for other prediction techniques as well. **Please do not worry about the technical specifics of every prediction technique (but we do recommend you to read about their basics if you have time)**. An application of the prediction command is that if we want to get to know more about a certain stock for investing purposes, we can study these models using different arguments within the models and draw conclusion on what model to use for the stock based on whichever model's prediction is closest to the real value (backtesting is an argument specifically for this).



3 CRYPTOCURRENCY

3.1 General Information

In order to access Cryptocurrency's data, we can simply type `crypto` on the terminal (figure below).

```
2022 May 15, 03:57 (Fri) / $ crypto
                                Cryptocurrency
load      load a specific cryptocurrency for analysis
find     find coins

Coin:
Source:

headlines  crypto sentiment from 15+ major news headlines [Finbrain]
chart      view a candle chart for a specific cryptocurrency
prt       potential returns tool - check how much upside if ETH reaches BTC market cap

> disc      discover trending cryptocurrencies, e.g.: top gainers, losers, top sentiment
> ov       overview of the cryptocurrencies, e.g.: market cap, DeFi, latest news, top exchanges, stab
> onchain  information on different blockchains, e.g.: eth gas fees, whale alerts, DEXes info
> defi     decentralized finance information, e.g.: dpi, llama, tvl, lending, borrow, funding
> tools    explore different tools e.g.: apytoapr, il
> nft      non-fungible tokens, e.g.: today drops
> dd       due-diligence for loaded coin, e.g.: coin information, social media, market stats
> ta       technical analysis for loaded coin, e.g.: ema, macd, rsi, adx, bbands, obv
> pred    prediction techniques, e.g.: regression, arima, rnn, lstm, convid, monte carlo
> qa       quantitative analysis e.g.: decompose, cusum, residuals analysis

                                         OpenBB Terminal v1.3.0m (https://openbb.co)
2022 May 15, 04:06 (Fri) /crypto/ $ -
```

Let's load the crypto's data by type `load` on the Terminal. For this situation, say we want to illustrate the analysis of *Ethereum (ETH)*. It will show basic information about the loaded crypto.

3.2 Digging Deeper

disc (discover trending cryptocurrency)

We use `disc` to discover the trending cryptocurrency in the market at the time we are using the terminal. To use the function, let's type `disc` on the Terminal.



```
2022 May 15, 04:20 (Fri) /crypto/ $ disc                               Cryptocurrency - Discovery
[CoinGecko]
  cgtop      top coins (with or without category)
  cgtranding   trending coins
  cggainers    top gainers - coins which price gained the most in given period
  cglosers     top losers - coins which price dropped the most in given period
[CoinPaprika]
  cpsearch    search for coins
[CoinMarketCap]
  cmctop      top coins
[DappRadar]
  drnft       top non fungible tokens
  drgames     top blockchain games
  drdapps     top decentralized apps
  drdex       top decentralized exchanges

OpenBB Terminal v1.3.0m (https://openbb.co)
```

The function `cgtop` is created to show the *top coins* (with or without categories) based on market capital.

- When we type `cgtop` in the terminal, it will automatically display the *top 10 cryptocurrency* that account for the largest market capital at the time we use the terminal.

```
2022 May 15, 04:22 (Fri) /crypto/disc/ $ cgtop
```

Symbol	Name	Volume [\$]	Market Cap [\$]	Market Cap Rank	7D Change [%]	24H Change [%]
btc	Bitcoin	23.7B	567.5B	1	-16.31	1.20
eth	Ethereum	13.2B	246B	2	-22.96	0.06
usdt	Tether	52.1B	75.9B	3	0.07	0.30
usdc	USD Coin	9.4B	51.2B	4	0.70	0.09
bnb	BNB	1.9B	49.3B	5	-19.92	2.25
xrp	XRP	2.2B	20.3B	6	-28.13	-2.36
ada	Cardano	877.1M	18B	7	-29.97	0.45
busd	Binance USD	13.5B	17.8B	8	0.47	0.16
sol	Solana	1.4B	17.4B	9	-34.99	1.49
dot	Polkadot	921.5M	12.1B	10	-20.74	4.20

- If you want to **categorize** the cryptos according to specific category, you can type `cgtop -c [YOUR CATEGORY]` or `cgtop --category [YOUR CATEGORY]`
e.g.: `cgtop -c stablecoins`



2022 May 15, 04:26 (UTC) /crypto/disc/ \$ cgtop -c stablecoins

Symbol	Name	Volume [\$]	Market Cap [\$]	Market Cap Rank	7D Change [%]	24H Change [%]
usdt	Tether	51.9B	75.9B	3.00	-0.04	0.30
usdc	USD Coin	9.4B	51.3B	4.00	0.58	0.07
busd	Binance USD	13.3B	17.8B	8.00	0.47	0.10
dai	Dai	631.9M	6B	17.00	0.12	0.22
ust	TerraUSD	896.8M	2.2B	35.00	-80.52	15.61
mim	Magic Internet Money	15.5M	1.8B	43.00	0.13	0.25
frax	Frax	41.9M	1.5B	51.00	-0.02	0.19
tusd	TrueUSD	108.1M	1.2B	59.00	0.10	0.14
usdp	Pax Dollar	11.8M	941.2M	71.00	0.21	0.24
usdn	Neutrino USD	32.6M	850.4M	76.00	-3.79	-0.15

- If you want to **sort** the cryptos according to *column index* (e.g.: Symbol, Name, etc.), type `cgtop -s [YOUR SORT]`. The indices have to be typed correctly, or else they will not work properly.

e.g.: `cgtop -s 7D Change [%]`

2022 May 15, 04:49 (UTC) /crypto/disc/ \$ cgtop -s 7D Change [%]

Symbol	Name	Volume [\$]	Market Cap [\$]	Market Cap Rank	7D Change [%]	24H Change [%]
usdc	USD Coin	9.4B	51.3B	4	0.57	-0.14
busd	Binance USD	13.3B	17.8B	8	0.48	0.06
usdt	Tether	52B	75.9B	3	-0.09	0.25
btc	Bitcoin	23.7B	566.3B	1	-16.41	0.81
bnb	BNB	1.9B	49.3B	5	-19.98	1.98
dot	Polkadot	974.5M	12.2B	10	-20.37	4.06
eth	Ethereum	13.1B	245.8B	2	-22.99	-0.11
xrp	XRP	2.4B	20.4B	6	-27.63	-1.94
ada	Cardano	879.2M	18.2B	7	-29.44	1.31
sol	Solana	1.4B	17.5B	9	-34.64	1.82

- If you want to **limit** the number of cryptos, type `cgtop -i [YOUR LIMIT]` or `cgtop --limit [YOUR LIMIT]`. The *default limit* is **10**; if you do not set the limit, it will automatically display the top 10 cryptos based on your category.

e.g.: `cgtop -i 50`



2022 May 15, 05:10 (UTC) /crypto/disc/ \$ cgtop -l 25

Symbol	Name	Volume [\$]	Market Cap [\$]	Market Cap Rank	7D Change [%]	24H Change [%]
btc	Bitcoin	23.7B	565B	1	-16.59	0.49
eth	Ethereum	13.2B	245B	2	-23.19	-0.94
usdt	Tether	51.7B	76B	3	0.15	0.40
usdc	USD Coin	9.2B	51.3B	4	0.39	-0.33
bnb	BNB	1.9B	49.1B	5	-20.19	1.16
xrp	XRP	2.3B	20.4B	6	-27.98	-2.93
ada	Cardano	871.6M	18.1B	7	-29.93	-0.20
busd	Binance USD	13.2B	17.8B	8	0.58	0.11
sol	Solana	1.4B	17.3B	9	-35.45	0.18
dot	Polkadot	977.6M	12.2B	10	-19.96	3.17
doge	Dogecoin	617.9M	11.7B	11	-31.28	-2.69
avax	Avalanche	611.5M	8.9B	12	-39.92	0.92

- If you want to export your data into either .csv, .json or .xlsx file, type `cgtop --export [YOUR TYPE OF FILE]`

e.g.: `cgtop --export csv`

2022 May 15, 05:13 (UTC) /crypto/disc/ \$ cgtop --export csv

Symbol	Name	Volume [\$]	Market Cap [\$]	Market Cap Rank	7D Change [%]	24H Change [%]
btc	Bitcoin	23.6B	565B	1	-16.74	0.14
eth	Ethereum	13.2B	245B	2	-23.25	-1.02
usdt	Tether	51.5B	76B	3	-0.22	-0.11
usdc	USD Coin	9.1B	51.1B	4	0.30	-0.42
bnb	BNB	1.9B	49.1B	5	-20.35	0.96
xrp	XRP	2.2B	20.4B	6	-28.18	-3.19
ada	Cardano	853.5M	18.1B	7	-30.08	-0.56
busd	Binance USD	13.2B	17.8B	8	0.54	0.07
sol	Solana	1.4B	17.2B	9	-35.64	-0.10
dot	Polkadot	976.2M	12.3B	10	-20.40	2.61

Saved file: C:/Users/aki/Desktop/OPENBB-exports/20220515_181359_cryptocurrency_disccovery_cgtop.csv



A	B	C	D	E	F	G	H	I
1	Symbol	Name	Volume [\$]	Market Cap [\$]	Market Cap Rank	7D Change [%]	24H Change [%]	
2	0 btc	Bitcoin	23.6B	565B	1	-16.74401367	0.138253797	
3	1 eth	Ethereum	13.2B	245B	2	-23.25362694	-1.019808571	
4	2 usdt	Tether	51.5B	76B	3	-0.219790334	-0.107016767	
5	3 usdc	USD Coin	9.1B	51.1B	4	0.304908295	-0.418202871	
6	4 bnb	BNB	1.9B	49.1B	5	-20.34901496	0.958938365	
7	5 xrp	XRP	2.2B	20.4B	6	-28.17980932	-3.190397249	
8	6 ada	Cardano	853.5M	18.1B	7	-30.08115793	-0.55908619	
9	7 busd	Binance USD	13.2B	17.8B	8	0.539631823	0.069369964	
10	8 sol	Solana	1.4B	17.2B	9	-35.63566646	-0.102536784	
11	9 dot	Polkadot	976.2M	12.3B	10	-20.40073892	2.607391677	
12								

It will also show the place it stores your data file, following the given directory.



4 FOREX

4.1. General Information

Other than the two *stocks* and *crypto* functions mentioned above, another set of data we might as well discover is *forex*. From the Home terminal, simply type **forex** as follows (the command **/ \$** is already there).

```
Home

Multiple jobs queue (where each '/' denotes a new command).
E.g. '/stocks $ disc/ugs -n 3/../load tsla/candle'

If you want to jump from crypto/ta to stocks you can use an absolute path that starts with a slash (/).
E.g. '/crypto/ta $ /stocks'

The previous logic also holds for when launching the terminal.
E.g. '$ python terminal.py /stocks/disc/ugs -n 3/../load tsla/candle'

You can run a standalone .openbb routine file with:
E.g. '$ python terminal.py routines/example.openbb'

You can run a .openbb routine file with variable inputs:
E.g. '$ python terminal.py routines/example_with_inputs.gst --input pltr,tsla,noi'

The main commands you should be aware when navigating through the terminal are:
cls           clear the screen
help / h / ?  help menu
quit / q / .. quit this menu and go one menu above
exit          exit the terminal
reset / r     reset the terminal and reload configs from the current location
resources      only available on main contexts (not sub-menus)

about          about us
update         update terminal automatically
tz             set different timezone
export          select export folder to output data
exe            execute automated routine script
> settings    set feature flags and style charts
> keys         set API keys and check their validity

Export Folder: DEFAULT (folder: exports/)
Timezone:      US/Eastern

> stocks
> crypto
> etf
> economy
> forex
> funds
> alternative
> econometrics
> portfolio
> jupyter

2022 May 16, 00:10 (UTC) / $ forex
```

OpenBB Terminal

In order to delve into the data of a specific foreign exchange market (Forex), one must first define the currencies in the forex pair. For example, let's say we want to find out the USD-JPY forex pair, so we would have to do the following steps.



- To define the base currency (which in this case is US Dollars), we type `from USD` as below.

```
Forex
from      select the "from" currency in a forex pair
to       select the "to" currency in a forex pair

From:  USD
To:
Source: YahooFinance

quote      get last quote [AlphaVantage]
load       get historical data
candle     show candle plot for loaded data

> ta        technical analysis for loaded coin,          e.g.: ema, macd, rsi, adx, bbands, obv
> qa        quantitative analysis,                   e.g.: decompose, cusum, residuals analysis
> pred     prediction techniques                  e.g.: regression, arima, rnn, lstm, convid, monte carlo

Forex brokerages:
> oanda    Oanda menu

2022 May 16, 00:58 (Fri) /forex/ $ from USD
Selected pair
From:  USD
To:
Source: YahooFinance
```

OpenBB Terminal

- To define the destination currency (which in this case is Japanese Yen), we type `to JPY` as below.

```
2022 May 16, 00:58 (Fri) /forex/ $ to JPY

Selected pair
From:  USD
To:    JPY
Source: YahooFinance
```

- In order to get data for later use, we ought to select a specific source to analyze. Although OpenBB Terminal will use *YahooFinance* source by default, we could also select other sources if we wish to have further objective market perspectives. The available sources are `yf` (*YahooFinance*), `av` (*AlphaAdvantage*), `oanda` (*OANDA*). However, within the `load` function, it does not merely comprise of historical data source. Let's take a detailed look through the `-h` flag, which will show the help dialogue below.



```
2022 May 16, 01:10 (UTC) /forex/ $ load -h
usage: load [--source {yf,av}] [-r {i,d,w,m}] [-i {1min,2min,5min,15min,30min,60min,1hour,1day,5day,1week,1month,3month}] [-s START_DATE] [-h]

Load historical exchange rate data. Available data sources are Alpha Advantage and YahooFinanceBy default main source used for analysis is YahooFinance (yf). To change it use --source av

optional arguments:
  --source {yf,av}      Source of historical data (default: yf)
  -r {i,d,w,m}, --resolution {i,d,w,m}
                        Resolution of data. Can be intraday, daily, weekly or monthly (default: d)
  -i {1min,2min,5min,15min,30min,60min,90min,1hour,1day,5day,1week,1month,3month}, --interval {1min,2min,5min,15min,30min,60min,90min,1hour,1day,5day,1week,1month,3month}
                        Interval of intraday data. Options: [YahooFinance] 1min, 2min, 5min, 15min, 30min, 60min, 90min, 1hour, 1day, 5day, 1week, 1month, 3month. [AlphaAdvantage] 1min, 5min, 15min, 30min, 60min (default: 5min)
  -s START_DATE, --start_date START_DATE
                        Start date of data. (default: 2022-03-18 14:45:16.210134)
  -h, --help            show this help message (default: False)
```

+ Let's load historical data on a weekly resolution for the last 1 month.

```
2022 May 16, 01:52 (UTC) /forex/ $ load --source yf -r w -s 2022-04-15

Selected pair
From: USD
To: JPY
Source: YahooFinance
```

4.2. Digging Deeper

a. General outlook

As we have the historical data loaded, we are able to retrieve real-time exchange rate.

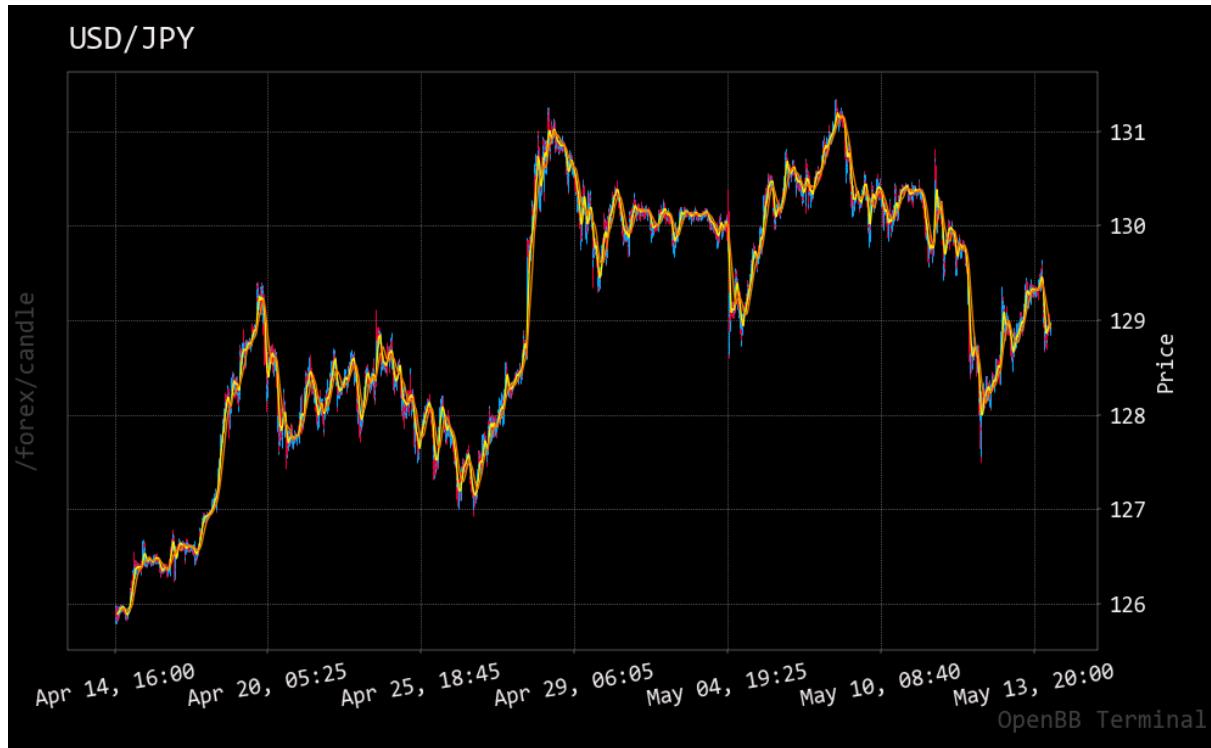
To do so, we use `quote` function as follows.

```
2022 May 16, 01:53 (UTC) /forex/ $ quote
USD/JPY Quote
```

Realtime Currency Exchange Rate	
From_Currency Code	USD
To_Currency Code	JPY
Last Refreshed	2022-05-16 06:17:43
Exchange Rate	128.97000000
Bid Price	128.97000000
Ask Price	128.97000000



If we use the `candle` function, a chart will appear in which it shows the fluctuation in currency exchange rates throughout the selected period of time (1 month within Apr 14 – May 13, 2022 in this case).



b. Detailed features

With a currency pair loaded, features from Technical Analysis menu are available. To discover what kind of the features are included, type `ta` as follows.

```
2022 May 16, 02:20 (Fri) /forex/ $ ta
Forex - Technical Analysis -
```

Currency Pair Loaded: USD/JPY (from 2022-04-14)
Source: YahooFinance

Overlap:

- ema exponential moving average
- sma simple moving average
- zlma zero lag moving average

Momentum:

- cci commodity channel index
- macd moving average convergence/divergence
- rsi relative strength index
- stoch stochastic oscillator
- fisher fisher transform
- cg centre of gravity

Trend:

- adx average directional movement index
- aroon aroon indicator

Volatility:

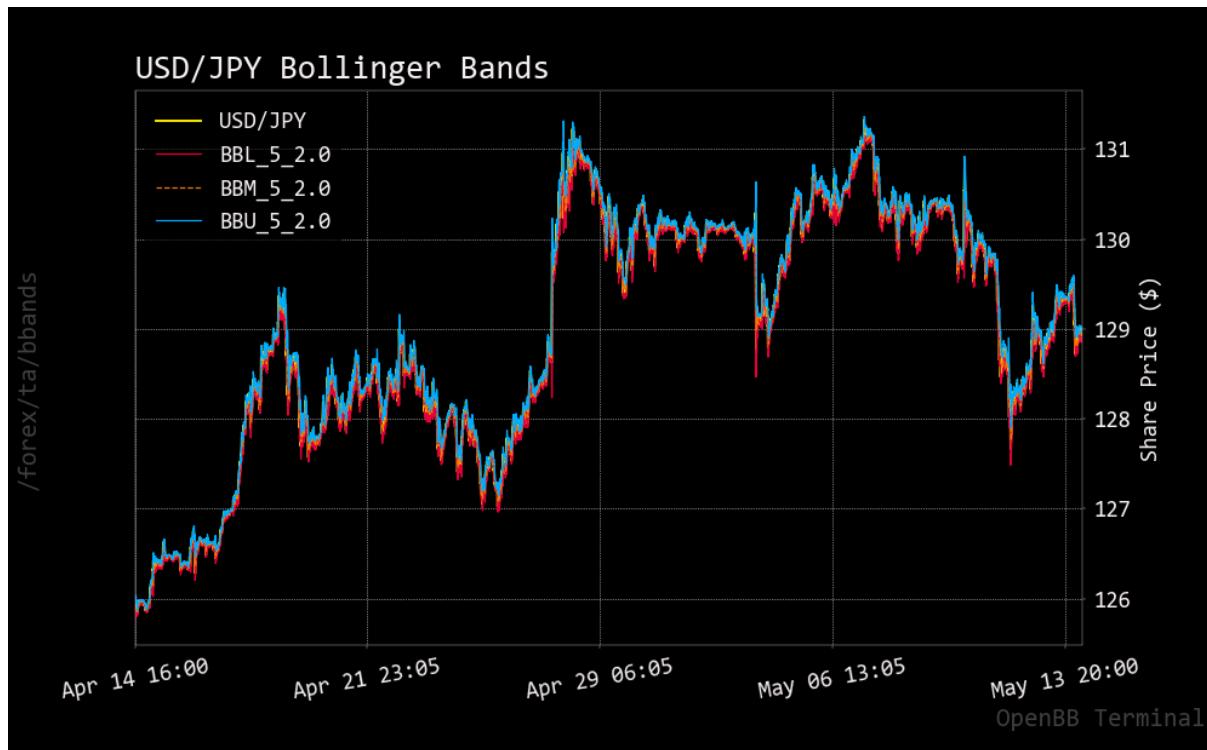
- bbands bollinger bands
- donchian donchian channels

Custom:

- fib fibonacci retracement



For example, let's say we want to examine the Bollinger Bands for USD/JPY forex pair. Bollinger Bands are envelopes plotted at a standard deviation level above and below a simple moving average of the price. Because the distance of the bands is based on standard deviation, they adjust to volatility swings in the underlying price. Bollinger Bands use 2 parameters, Period and Standard Deviations (StdDev). The default values are 20 for period, and 2 for standard deviations, although one could customize the combinations. In OpenBB Terminal, we can use the `bbands` feature under the `Volatility` category.





5 ECONOMETRICS

5.1. General Information

Econometrics is basically using mathematics (particularly statistics) to analyze *economic trends, patterns*, and test different *economic hypotheses*. In other words, econometrics is the statistical analysis of data in order to test or advance economic theory, quantify and understand economic ideas. The goal of OpenBB Terminal's *econometrics* menu is to allow the user to do statistical research on customized datasets. The menu allows the user to import their own dataset(s), alter the data by (for example) adding columns or adjusting indexes, and perform statistical tests.

Let's try loading some econometrics commands (figure below). We can simply type `econometrics` from home to get this.

```
[2022 May 22, 06:38 (🦋) / $ econometrics
Econometrics ━━━━━━━━
load      load in custom data sets
export    export a dataset
remove   remove a dataset
options   show available column-dataset options

Loaded files: None

Exploration
show      show a portion of a loaded dataset
plot      plot data from a dataset
type      change types of the columns or display their types
desc      show descriptive statistics of a dataset
index    set (multi) index based on columns
clean     clean a dataset by filling or dropping NaNs
modify   combine columns of datasets and delete or rename columns

Timeseries
ols       fit a (multi) linear regression model
norm     perform normality tests on a column of a dataset
root      perform unitroot tests (ADF & KPSS) on a column of a dataset

Panel Data
panel    estimate model based on various regression techniques
compare  compare results of all estimated models

Tests
dwat     Durbin-Watson autocorrelation test on the residuals of the regression
bgod     Breusch-Godfrey autocorrelation tests with lags on the residuals of the regres
bpag     Breusch-Pagan heteroscedasticity test on the residuals of the regression
granger  Granger causality tests on two columns
coint    co-integration test on a multitude of columns

OpenBB Terminal
2022 May 22, 06:38 (🦋) /econometrics/ $ ]
```



Let's start from the fundamentals. We want to see what we can do with the `load` command. Following are the different arguments we can make use of within the `load` command.

```
[2022 May 22, 06:50 (🦋) /econometrics/ $ load -h
usage: load [-f FILE [FILE ...]] [-ex] [-h]

Load custom data set into a dataframe

optional arguments:
  -f FILE [FILE ...], --file FILE [FILE ...]
                        File to load in and the alias you wish to give to the dataset (default:
                        None)
  -ex, --examples      Use this argument to show examples of Statsmodels to load in. See:
                        https://www.statsmodels.org/devel/datasets/index.html (default: False)
  -h, --help            show this help message (default: False)

2022 May 22, 06:50 (🦋) /econometrics/ $ █
```

This is mainly for loading custom datasets that we have prepared ourselves. Additionally, we can also make use of the available example datasets within the terminal using this command (more about this below). In addition, the commands `export`, `remove`, and `options` all complement the `load` command and we can make use of them in a similar fashion to how we have used similar commands so far.

5.2. Digging Deeper

Let's use some example datasets to explore the `econometrics` menu. We can do that by the optional argument `-ex` within the `load` command (discovered by the `-h` flag).

```
[2022 May 22, 06:50 (🦋) /econometrics/ $ load -h
usage: load [-f FILE [FILE ...]] [-ex] [-h]

Load custom data set into a dataframe

optional arguments:
  -f FILE [FILE ...], --file FILE [FILE ...]
                        File to load in and the alias you wish to give to the dataset (default:
                        None)
  -ex, --examples      Use this argument to show examples of Statsmodels to load in. See:
                        https://www.statsmodels.org/devel/datasets/index.html (default: False)
  -h, --help            show this help message (default: False)

2022 May 22, 06:50 (🦋) /econometrics/ $ █
```

This appears on the terminal when we enter the `-ex` argument.



```
[2022 May 22, 06:50 (🦋) /econometrics/ $ load -ex
          0
anes96           American National Election Survey 1996
cancer           Breast Cancer Data
ccard            Bill Greene's credit scoring data.
cancer_china    Smoking and lung cancer in eight cities in China.
co2              Mauna Loa Weekly Atmospheric CO2 Data
committee        First 100 days of the US House of Representatives 1995
copper           World Copper Market 1951–1975 Dataset
cpunish          US Capital Punishment dataset.
danish_data     Danish Money Demand Data
elnino           El Nino – Sea Surface Temperatures
engel            Engel (1857) food expenditure data
fair              Affairs dataset
fertility        World Bank Fertility Data
grunfeld         Grunfeld (1950) Investment Data
heart             Transplant Survival Data
interest_inflation (West) German interest and inflation rate 1972–1998
longley          Longley dataset
macrodata        United States Macroeconomic data
modechoice       Travel Mode Choice
nile              Nile River flows at Ashwan 1871–1970
randhie          RAND Health Insurance Experiment Data
scotland         Taxation Powers Vote for the Scottish Parliament 1997
spector          Spector and Mazzeo (1980) – Program Effectiveness Data
stackloss         Stack loss data
star98           Star98 Educational Dataset
statecrim        Statewide Crime Data 2009
strikes          U.S. Strike Duration Data
sunspots         Yearly sunspots data 1700–2008
wage_panel       Veila and M. Verbeek (1998): Whose Wages Do Unions Raise?
2022 May 22, 06:53 (🦋) /econometrics/ $
```

Let's suppose we want to analyze the United States macroeconomic data in order to test or advance some economic theory of ours. We load the macroeconomic data (figure below).

```
2022 May 22, 06:58 (🦋) /econometrics/ $ load macrodata database
2022 May 22, 06:58 (🦋) /econometrics/ $ show
      year quarter realgdp realcons realinv realgovt realdipi cpi m1 tbilrate unemp pop infl realint
0 1959.0 1.0 2710.349 1707.4 286.898 470.045 1886.9 28.98 139.7 2.82 5.8 177.146 0.00 0.00
1 1959.0 2.0 2778.801 1733.7 310.859 481.301 1919.7 29.15 141.7 3.08 5.1 177.830 2.34 0.74
2 1959.0 3.0 2775.488 1751.8 289.226 491.260 1916.4 29.35 140.5 3.82 5.3 178.657 2.74 1.09
3 1959.0 4.0 2785.204 1753.7 299.356 484.052 1931.3 29.37 140.0 4.33 5.6 179.386 0.27 4.06
4 1960.0 1.0 2847.699 1770.5 331.722 462.199 1955.5 29.54 139.6 3.50 5.2 180.007 2.31 1.19
5 1960.0 2.0 2834.390 1792.9 298.152 460.400 1966.1 29.55 140.2 2.68 5.2 180.671 0.14 2.55
6 1960.0 3.0 2839.022 1785.8 296.375 474.676 1967.8 29.75 140.9 2.36 5.6 181.528 2.70 -0.34
7 1960.0 4.0 2802.616 1788.2 259.764 476.434 1966.6 29.84 141.1 2.29 6.3 182.287 1.21 1.08
8 1961.0 1.0 2819.264 1787.7 266.405 475.854 1984.5 29.81 142.1 2.37 6.8 182.992 -0.40 2.77
9 1961.0 2.0 2872.005 1814.3 286.246 480.328 2014.4 29.92 142.9 2.29 7.0 183.691 1.47 0.81

      year quarter realgdp realcons realinv realgovt realdipi cpi m1 tbilrate unemp pop infl realint
0 1959.0 1.0 2710.349 1707.4 286.898 470.045 1886.9 28.98 139.7 2.82 5.8 177.146 0.00 0.00
1 1959.0 2.0 2778.801 1733.7 310.859 481.301 1919.7 29.15 141.7 3.08 5.1 177.830 2.34 0.74
2 1959.0 3.0 2775.488 1751.8 289.226 491.260 1916.4 29.35 140.5 3.82 5.3 178.657 2.74 1.09
3 1959.0 4.0 2785.204 1753.7 299.356 484.052 1931.3 29.37 140.0 4.33 5.6 179.386 0.27 4.06
4 1960.0 1.0 2847.699 1770.5 331.722 462.199 1955.5 29.54 139.6 3.50 5.2 180.007 2.31 1.19
5 1960.0 2.0 2834.390 1792.9 298.152 460.400 1966.1 29.55 140.2 2.68 5.2 180.671 0.14 2.55
6 1960.0 3.0 2839.022 1785.8 296.375 474.676 1967.8 29.75 140.9 2.36 5.6 181.528 2.70 -0.34
7 1960.0 4.0 2802.616 1788.2 259.764 476.434 1966.6 29.84 141.1 2.29 6.3 182.287 1.21 1.08
8 1961.0 1.0 2819.264 1787.7 266.405 475.854 1984.5 29.81 142.1 2.37 6.8 182.992 -0.40 2.77
9 1961.0 2.0 2872.005 1814.3 286.246 480.328 2014.4 29.92 142.9 2.29 7.0 183.691 1.47 0.81

2022 May 22, 06:58 (🦋) /econometrics/ $
```



Here, we first loaded the macroeconomic data by using `load microdata database` and used the `show` command to show a random portion of the available data frame.

We can also make use of different optional arguments to manipulate the available data to our preferred utilization (figure below).

```
[2022 May 22, 07:00 (W) /econometrics/ $ show -h
usage: show [-n {dataset,database}] [-s SORTCOL [SORTCOL ...]] [-a] [-h] [--export EXPORT] [-l LIMIT]

Show a portion of the DataFrame

optional arguments:
  -n {dataset,database}, --name {dataset,database}
                                The name of the database you want to show data for (default: None)
  -s SORTCOL [SORTCOL ...], --sortcol SORTCOL [SORTCOL ...]
                                Sort based on a column in the DataFrame (default: )
  -a, --ascend               Use this argument to sort in a descending order (default: False)
  -h, --help                  show this help message (default: False)
  --export EXPORT             Export raw data into csv, json, xlsx (default: )
  -l LIMIT, --limit LIMIT
                                Number of entries to show in data. (default: 10)

2022 May 22, 07:00 (W) /econometrics/ $ ]
```

Let's try to get some plots. We can load the `help` argument within the `plot` command (figure below) to get the idea of what we can do.

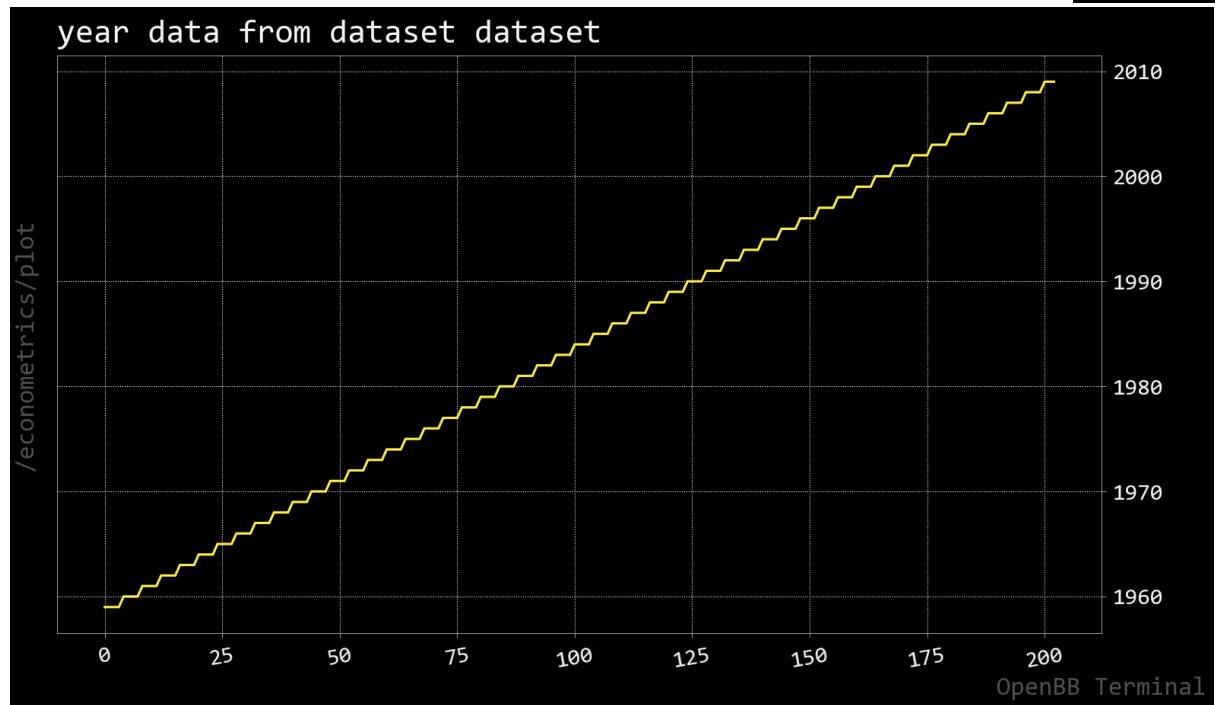
```
[2022 May 22, 07:15 (W) /econometrics/ $ plot -h
usage: plot
  [-c {year-dataset,quarter-dataset,realgdp-dataset,realcons-dataset,realinv-dataset,realgovt-dataset,realdpi-dataset,cpi-dataset,m1-dataset,tbilarate-dataset,unemp-dataset,pop-dataset,infl-dataset,realint-dataset,year-database,quarter-database,realgdp-database,realcons-database,realinv-database,realgovt-database,realdpi-database,cpi-database,m1-database,tbilarate-database,unemp-database,pop-database,infl-database,realint-database}]
  [-h] [--export EXPORT]

Plot data based on the index

optional arguments:
  -c {year-dataset,quarter-dataset,realgdp-dataset,realcons-dataset,realinv-dataset,realgovt-dataset,realdpi-dataset,cpi-dataset,m1-dataset,tbilarate-dataset,unemp-dataset,pop-dataset,infl-dataset,realint-dataset,year-database,quarter-database,realgdp-database,realcons-database,realinv-database,realgovt-database,realdpi-database,cpi-database,m1-database,tbilarate-database,unemp-database,pop-database,infl-database,realint-database}
                                Column to plot along the index (default: None)
  -h, --help                  show this help message (default: False)
  --export EXPORT             Export figure into png, jpg, pdf, svg (default: )

2022 May 22, 07:15 (W) /econometrics/ $ ]
```

This suggests that we can load charts of different data such in terms of `year`, `quarter`, `realgdp`, `cpi`, and so on, using the `-c (column)` argument. If we load `plot -c year-dataset`, the following chart pops up:



We can export this figure as well using the `--export` argument.



APPENDIX

Setup

For Windows users:

- Go to: <https://github.com/OpenBB-finance/OpenBTerminal/releases>
- Check on the latest OpenBB Terminal version

The screenshot shows a GitHub release page for 'OpenBB Terminal v1.3.0'. At the top right, there is a 'Latest' button with a red box around it. Below the button, a message says 'A big thank you and welcome to our new contributors:' followed by a list of names (@josephjacks, @DidierRLoPES, etc.) and small heart icons. Underneath, a section titled 'What's changed:' lists numerous commits from May 2022, including fixes for signatures, reports, dashboards, and crypto features. At the bottom, there is a note about community contributions and a call to action to join the Discord server.

- Click on the “.exe” file

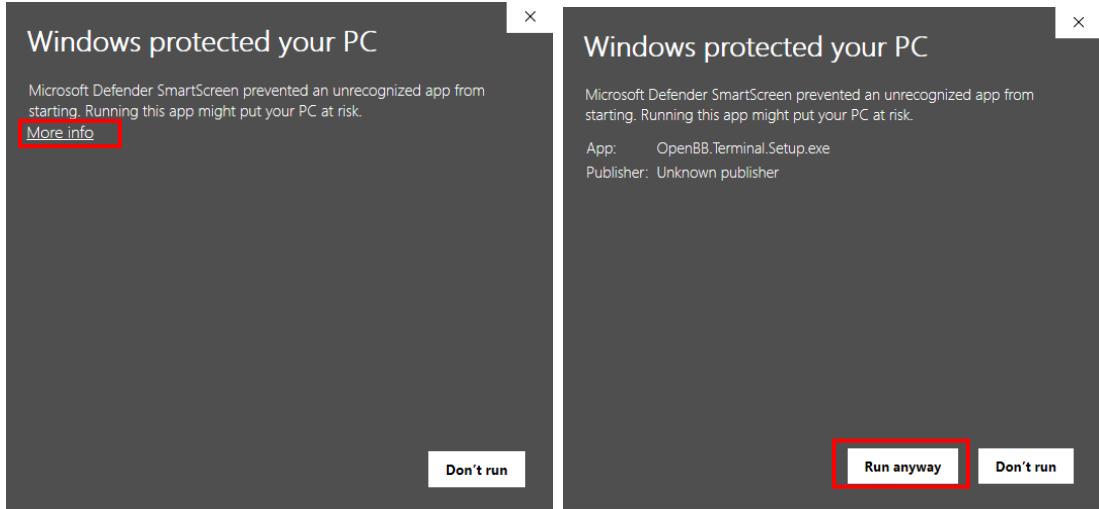
The screenshot shows the 'Assets' section of the GitHub release page. It lists four files: 'OpenBB.Terminal.dmg' (505 MB), 'OpenBB.Terminal.Setup.exe' (759 MB, highlighted with a red box), 'Source code (zip)', and 'Source code (tar.gz)'. Below the files, there are reaction counts: 2 thumbs up, 1 comment, and 3 people reacted.

- Double click the .exe file that got downloaded to your "Downloads" folder.
Note: You will most likely receive the error on the right stating "Windows protected your PC". This is because the installer is still in beta phase, and the team has not yet requested verification from Windows.

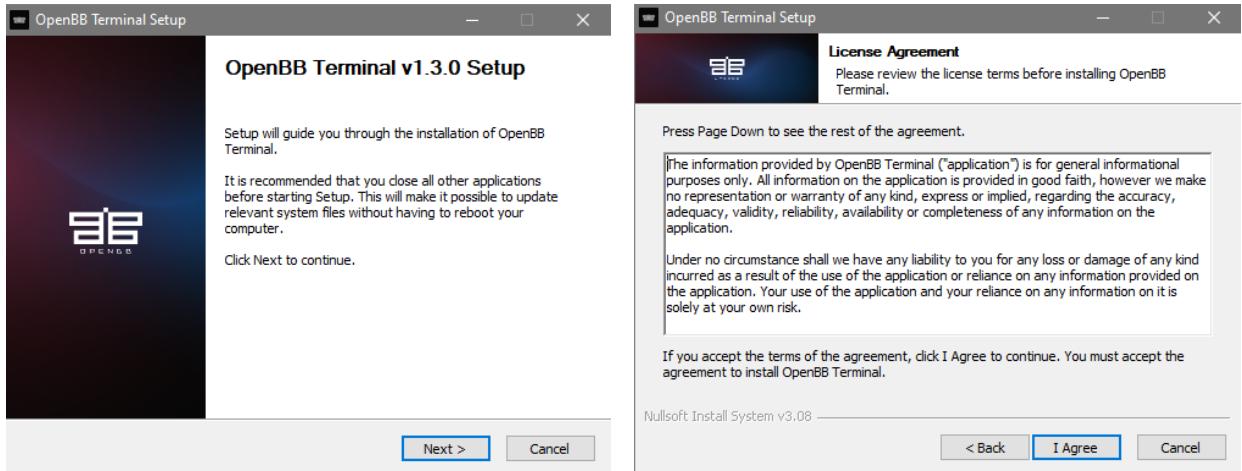


- Click on "More info" and select "Run anyway" to start the installation process. Proceed by following the steps.

Note: If you wish to install the application to 'Program Files' that you will have to run the resulting application as Administrator.



- After that, the window appears, click "Next" → "I agree" → Choose a place to install → "Finish"



- By clicking on the application that appeared on your Desktop, you are able to run the Terminal.

Note: The first time this takes a bit longer to load.

For Mac OS users:

- Go to: <https://github.com/OpenBB-finance/OpenBCTerminal/releases>
- Check on the latest OpenBB Terminal version:



3 days ago

piiq v1.3.0 e222859 Compare

OpenBB Terminal v1.3.0 Latest

A big thank you and welcome to our new contributors:

@josephjacks

What's changed:

2022-05-12 - OpenBB Terminal blended signature (#1806) by @DidierRLoPES
2022-05-12 - Reports and Dashboards to be one level above (#1805) by @DidierRLoPES
2022-05-12 - fix 1756 (#1801) by @jose-donato
2022-05-12 - Improve overall UX (#1804) by @DidierRLoPES
2022-05-11 - fix 1793; bring stocks ta to crypto (#1802) by @jose-donato
2022-05-12 - Fix regression bug introduced by #1709 regarding support command (#1803) by @DidierRLoPES
2022-05-12 - Fixes 1636 - hugo docs showing white bg (#1800) by @jose-donato
2022-05-12 - Add /crypto/ta/kc implementation (#1799) by @alexferrari88
2022-05-11 - Trading hours stock feature (#1697) by @buahaha
2022-05-11 - Improve error keys for reddit and twitter error (#1797) by @DidierRLoPES
2022-05-11 - Fix crypto loading for Binance #1794 (#1795) by @alexferrari88
2022-05-11 - change view_vwap to specify date range (#1790) by @PzaThief
2022-05-10 - Update README - Very minor grammatical typos (#1792) by @josephjacks ✨
2022-05-10 - Refactored Crypto Tests (#1743) by @colin99d
2022-05-11 - Add terminal-wide report command (#1709) by @minhhoang1023

We're proud of our community contributors and staunch supporters of open-source ecosystems.
Help us promote our community by tagging `@openbb_finance` on Twitter with a link to your pull request, and join our Discord server to chat about your contribution! We want to hear about your experience!

- Click on the “.dmg” file

▼ Assets 4

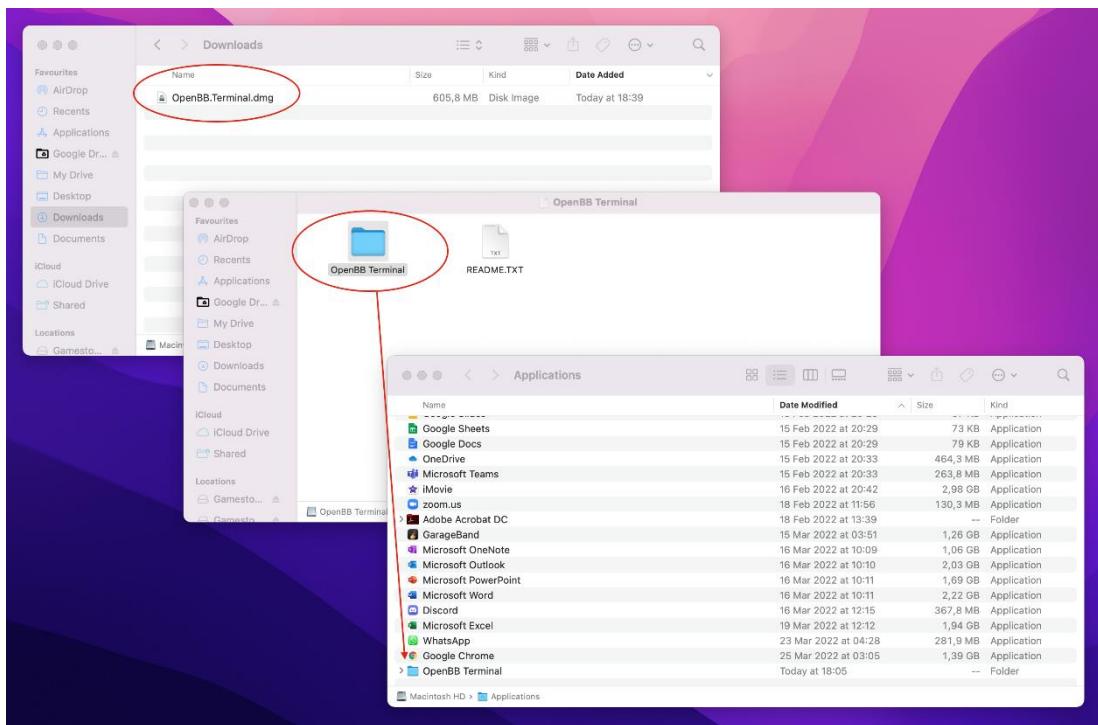
OpenBB.Terminal.dmg	505 MB
OpenBB.Terminal.Setup.exe	759 MB
Source code (zip)	
Source code (tar.gz)	

2 1 3 people reacted

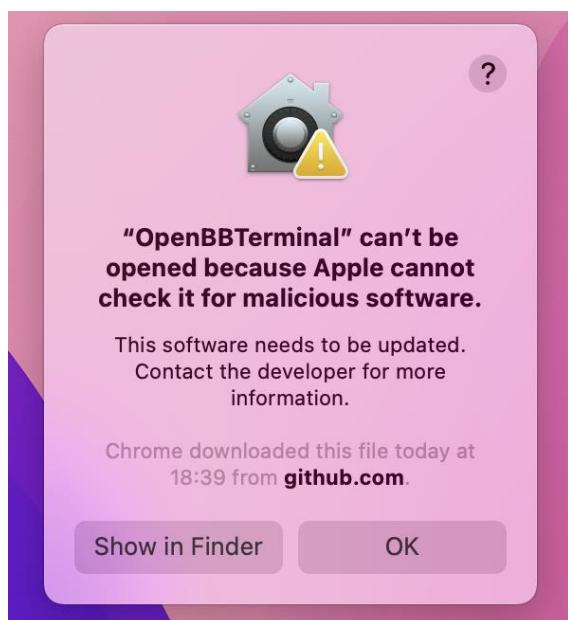
- Open the .dmg file that got downloaded to your "Downloads" folder.
- Drag the folder into your "Applications" folder. This should take some time as it is extracting the files from the .dmg file.



- Open the folder and click on the "OpenBB Terminal" file, this should run the application.



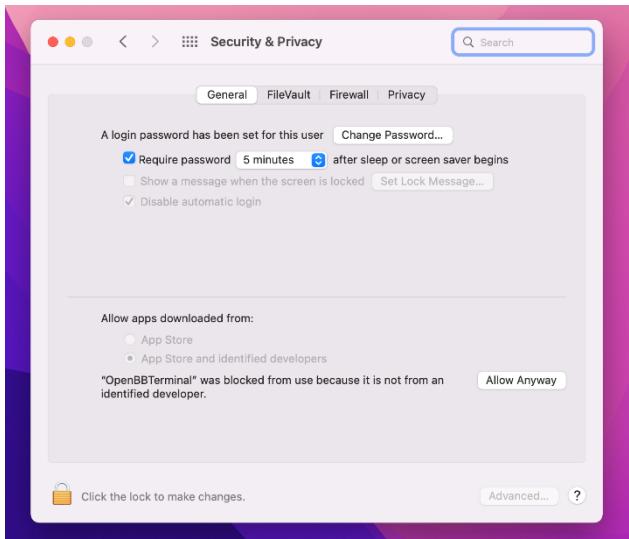
Notes:



It is possible you will see the image displayed below.

This is because the installer is still in beta phase, and the team has not yet requested verification from Apple. Please see below how to correct for this. There are two ways to fix this:

Option 1: via MacOS Settings

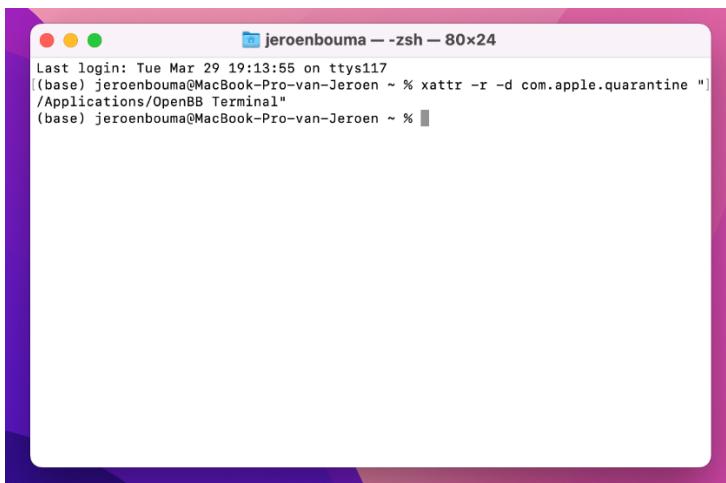


- Run the file. You will most likely receive a message that macOS was not able to check whether the application contains malicious software. The reasoning for this is described above.

- Go to System Preferences > Security & Privacy > General. You should see a message at the bottom that says that the file "was blocked from use because it is not from an identified developer". Click on "Allow anyway" or "Open anyway".
- If you receive the above message multiple times on different files, keep allowing the files via Step 2 or use Option 2.

- You should now be able to launch the application.

Option 2: via Terminal



- Run the file. You will most likely receive a message that macOS was not able to check whether the application contains malicious software. The reasoning for this is described in the note above.

- Open the "Terminal" application. This can be done by using Command (⌘) + Space and typing "terminal".
- Within this "Terminal" application, assuming you moved the file to the Applications folder,

enter the following command:

`xattr -r -d com.apple.quarantine "/Applications/OpenBB Terminal"`

- You should now be able to launch the application.



API KEYS

What is an API Key?

An application programming interface (API) key is a code that computer applications use to access the API in order to identify their user, developer, or calling programs to a website. API keys are often used to track and regulate how the interface is being used. This is frequently done to avoid misuse or malicious usage of the API in concern. An API key can serve as both a secret authentication token and a unique identification.

One of those calling programs is the OpenBB Terminal. The terminal actively makes use of API keys from different outlets to integrate them to its commands to access historical data or financials. Users need to enter the API keys to the terminal manually.

Updating API keys in the Terminal

We are able to directly set different variables from within the "keys" menu. This menu can be found on the first menu we see when opening the terminal (figure below).

```
Home

Multiple jobs queue (where each '/' denotes a new command).
E.g. '/stocks $ disc/ugs -n 3/../load tsla/candle'

If you want to jump from crypto/ta to stocks you can use an absolute path that starts with a slash (/).
E.g. '/crypto/ta $ /stocks'

The previous logic also holds for when launching the terminal.
E.g. '$ python terminal.py /stocks/disc/ugs -n 3/../load tsla/candle'

You can run a standalone .openbb routine file with:
E.g. '$ python terminal.py routines/example.openbb'

You can run a .openbb routine file with variable inputs:
E.g. '$ python terminal.py routines/example_with_inputs.gst --input pltr,tsla,mio'

The main commands you should be aware when navigating through the terminal are:
cls           clear the screen
help / h / ?  help menu
quit / q / .. quit this menu and go one menu above
exit          exit the terminal
reset / r     reset the terminal and reload configs from the current location
resources      only available on main contexts (not sub-menus)

about          about us
update         update terminal automatically
tz             set different timezone
export         select export folder to output data
exe            execute automated routine script
> settings    set feature flags and style charts
> keys        set API keys and check their validity
```

Then, by typing the command on the left-hand side, followed by our API key, we can set the key and use the commands attached to it (figure below).

```
Set API keys through environment variables:
av                      ALPHA_VANTAGE      defined, test passed
fmp                     FINANCIAL_MODELING_PREP defined, test passed
quandl                  QUANDL             defined, test passed
polygon                 POLYGON            defined, test passed
fred                     FRED               defined, test passed
news                     NEWSAPI           defined, test passed
tradier                 TRADIER            not defined
cmc                     COINMARKETCAP    not defined
finnhub                 FINNHUB            defined, test passed
iex                     IEXCLOUD           not defined
reddit                  REDDIT              not defined
twitter                 TWITTER            defined, test passed
rh                      ROBINHOOD          not defined
degiro                  DEGIRO              not defined
oanda                   OANDA              not defined
binance                 BINANCE             not defined
bitquery                BITQUERY            not defined
si                      SENTIMENT_INVESTOR not defined
cb                      COINBASE            not defined
wa                      WHALE_ALERT         not defined
glassnode               GLASSNODE          not defined
coinglass               COINGLASS           not defined
cpanic                  CRYPTO_PANIC        not defined
ethplorer               ETHPLORER          defined, test passed
smartstake              SMARTSTAKE          not defined
github                  GITHUB              not defined
```

2022 May 15, 04:55 (🦋) /keys/ \$ news YOUR_KEY_HERE █

But how do we get API Keys for OpenBB (or anything in general)?

We can get our individual API keys from any environment variable listed on the OpenBB Terminal. In order to know the website of a particular environment variable, we can simply type the command on the left-hand side (figure below).

```
Set API keys through environment variables:
av                      ALPHA_VANTAGE      defined, test passed
fmp                     FINANCIAL_MODELING_PREP defined, test passed
quandl                  QUANDL             defined, test passed
polygon                 POLYGON            defined, test passed
fred                     FRED               defined, test passed
news                     NEWSAPI           defined, test passed
tradier                 TRADIER            not defined
cmc                     COINMARKETCAP    not defined
finnhub                 FINNHUB            defined, test passed
iex                     IEXCLOUD          not defined
reddit                  REDDIT             not defined
twitter                 TWITTER            defined, test passed
rh                      ROBINHOOD         not defined
degiro                  DEGIRO             not defined
oanda                   OANDA              not defined
binance                 BINANCE            not defined
bitquery                BITQUERY           not defined
si                      SENTIMENT_INVESTOR not defined
cb                      COINBASE           not defined
wa                      WHALE_ALERT        not defined
glassnode               GLASSNODE          not defined
coinglass               COINGLASS          not defined
cpanic                  CRYPTO_PANIC      not defined
ethplorer               ETHPLORER          defined, test passed
smartstake              SMARTSTAKE         not defined
github                  GITHUB              not defined
```

```
[2022 May 15, 07:01 (🦋) /keys/ $ github
For your API Key, visit: https://docs.github.com/en/rest/guides/getting-started-with-the-rest-api
```

We should then visit the stated webpage (or we can also simply type **name + API Key** on Google; e.g., for **github**, simply type **github API Key**) and follow the instructions stated by the website. Every website mentioned on the terminal grants its users access to their own API keys, and these keys must be kept safe somewhere, since they will be useful for our projects more than just the OpenBB Terminal.

After we get the API key (or token), we can simply type the command on the left-hand side, followed by our API key (figure below). After the terminal responds with either ‘**defined**, **test passed**’ OR ‘**defined**’, we are set.

```
[2022 May 15, 07:01 (🦋) /keys/ $ github add_the_token_here_and_press_enter
defined
```

In order to verify this, we can load the keys once again (figure below). Our example environment variable ‘**GITHUB**’ is now defined (also confirm: **av**, **fmp**, **news**, **finnhub**, and others are listed as ‘**defined, test passed**’; same procedure had been used in order to set those API keys).



[2022 May 15, 07:05 (UTC) /keys/ \$ h

Keys

Set API keys through environment variables:		
av	ALPHA_VANTAGE	defined, test passed
fmp	FINANCIAL_MODELING_PREP	defined, test passed
quandl	QUANDL	defined, test passed
polygon	POLYGON	defined, test passed
fred	FRED	defined, test passed
news	NEWSAPI	defined, test passed
tradier	TRADIER	not defined
cmc	COINMARKETCAP	not defined
finnhub	FINNHUB	defined, test passed
iex	IEXCLOUD	not defined
reddit	REDDIT	not defined
twitter	TWITTER	defined, test passed
rh	ROBINHOOD	not defined
degiro	DEGIRO	not defined
oanda	OANDA	not defined
binance	BINANCE	not defined
bitquery	BITQUERY	not defined
si	SENTIMENT_INVESTOR	not defined
cb	COINBASE	not defined
wa	WHALE_ALERT	not defined
glassnode	GLASSNODE	not defined
coinglass	COINGLASS	not defined
cpanic	CRYPTO_PANIC	not defined
ethplorer	ETHPLORER	defined, test passed
smartstake	SMARTSTAKE	not defined
github	GITHUB	defined