StockViewer
An easy tool for stock buyers.
Front end : Eason
API: Tony
(then Eason made everything perfect)

Why we do this?

- Many people are attracted by finance product because it can make money.
- Hope to create a tool at website, people can check the stock’s status at a glance.

stonks

What Question does StockViewer answer?

Which Stocks
vary the most?

Demo Link:

https://easonc13.github.io/StockViewer/

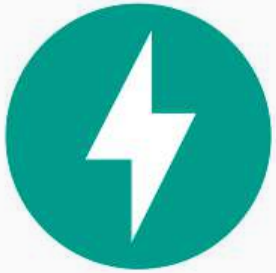
Visualization Overview

- We wanted a visualization that can more intuitively tell the standard deviations of each queried stock by dynamically via green for negative percent change, red for positive percent changes and the size of the bubble representing the magnitude of the change.
- Users can also reference API queried JSON when hovering over bubble on the left hand side
- Users can delete real time bar queries to be visualized

Data Flow

- Used The Alpaca Trade API To get real-time Stock info
- Calculate standard deviation by finding percent change of each stocks via Python.
- We feed JSON data via Fast API to front end by URL request.
- Visualize Data in Front-end

What is FastAPI?
(Say no to JavaScript!)



FastAPI

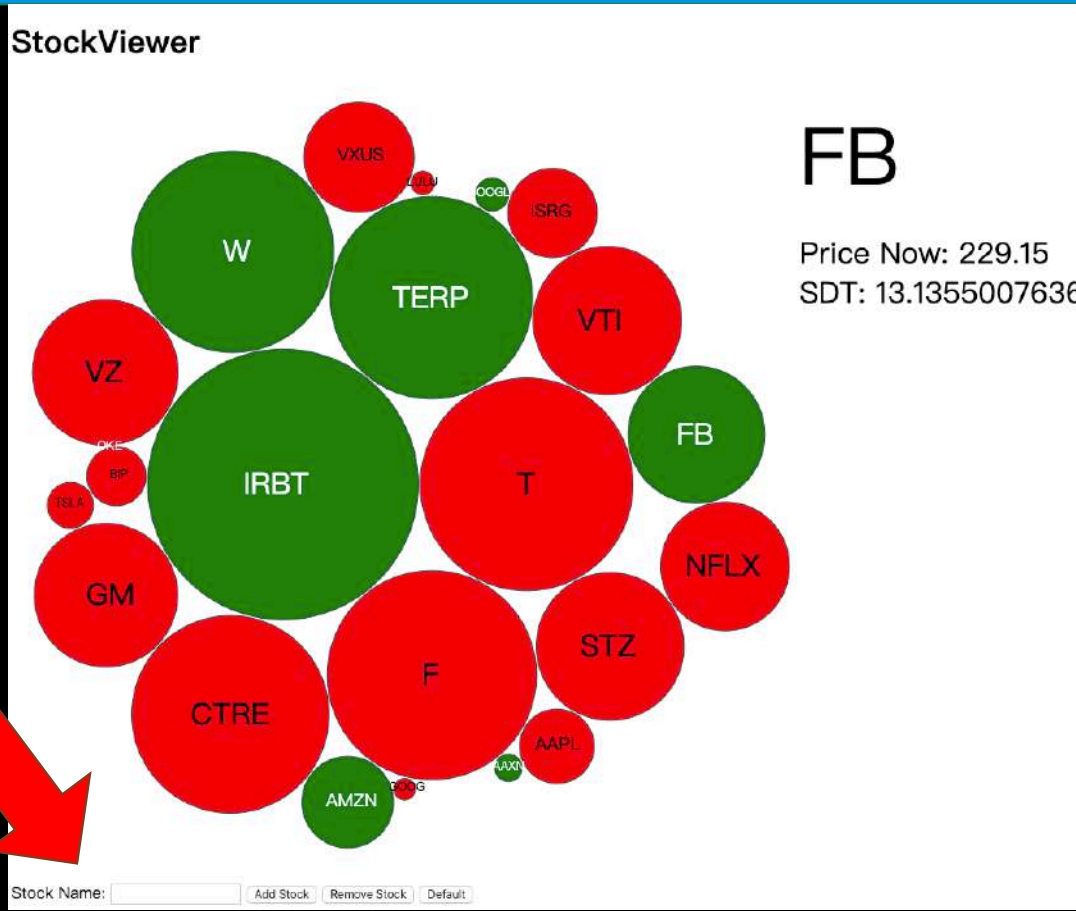
High performance, easy to learn,
fast to code, ready for production

Example Code

```python
import alpaca_trade_api as tradeapi

api_key = api_key
api_secret = api_secret
base_url = base_url

@app.post("/alpaca/{stock_id_list}/{timeframe}")
async def barrs(stock_id_list,timeframe,limit = 20):#token: str = Depends(oauth2_scheme)):
    api = tradeapi.REST(api_key,api_secret,base_url,api_version='v2')
    account = api.get_account()

    stock_id_list = stock_id_list.replace(">","").split("<")
    stock_id_list = [i.upper() for i in stock_id_list]

    if len(stock_id_list[1:]) > int(limit):
        return {"Query_Limit":"Exceeded Query Limit"}

    data = dict()
    for stock in stock_id_list[1:]:
        try:
            bar = api.get_barset(stock,timeframe,limit=2)
            STD = (bar[stock][1].c - bar[stock][0].c)/ bar[stock][1].c
        except IndexError as e:
            return {"Query Error":"one of your queried stocks does not exist"}
        except Exception as e:
            return {"Timeframe Error":{"please choose":{'minute','1Min','5Min','15Min','day','1D'}}}

        data[stock] = dict()
        data[stock]["ID"] = stock
        data[stock]["SDT"] = STD
        data[stock]["Price_Now"] = bar[stock][-1].c

    return data
```

Design of the Web Site

- D3 Create Bubble
- Mouse moving interactive
- Clicking to go Google Page.
- Add/Delete Stock below.
- Changes store in cookies.

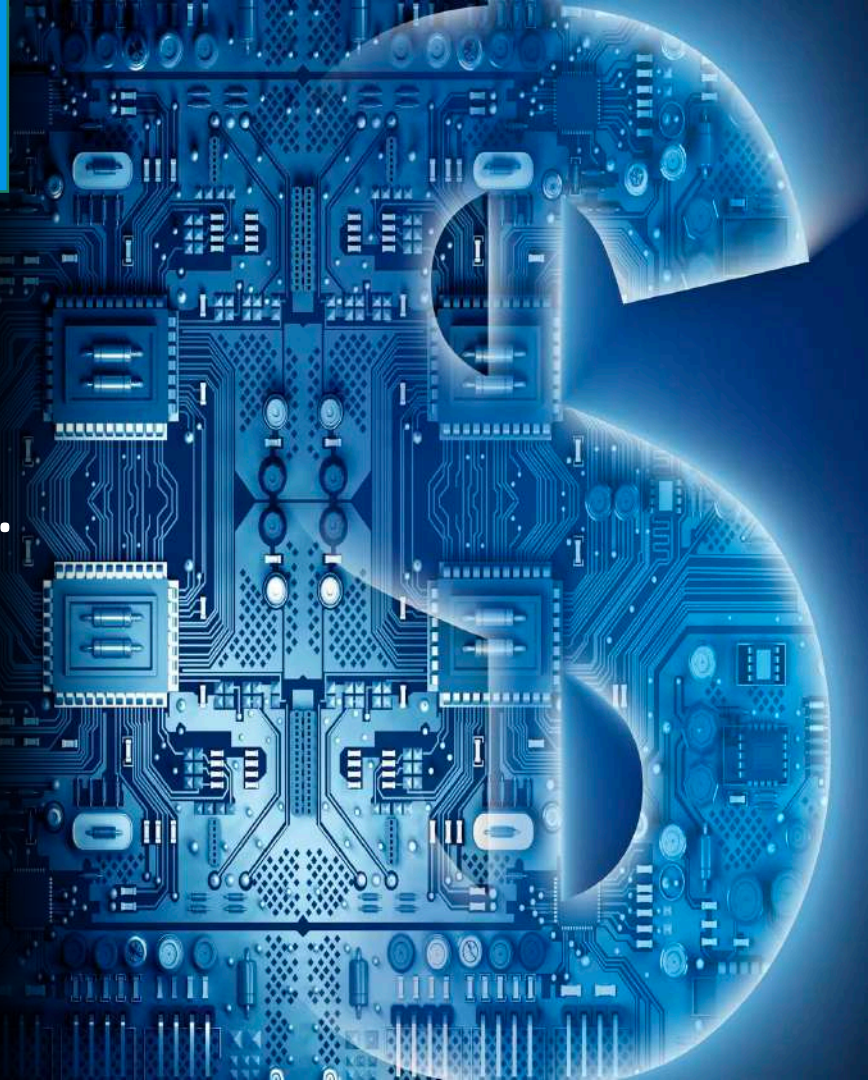


Demo Link:

https://easonc13.github.io/StockViewer/

Future

- Back-endless
- Allow more finance product.
  - TPE’s Stock.
  - Crypto Token.
- Handle more information.



EasonC13 / StockViewer

Pull Requests are welcome.
Feel free to fork on my GitHub.

You also can follow us on GitHub.
(EasonC13, Tony363)

https://github.com/EasonC13/StockViewer