Yash Jalan

CS-110 Final Project Report

My final project was based primarily on the idea of creating an Investment Advisory program. Its basic purpose was to inform the user on what the best investment would be purely based on the numbers provided by two key market metrics- the P/E ratio and the Dividend Yield. It would then also graph the stock it chose, followed by the S and P 500. This would show the user a visual representation on how the stock chosen is doing compared to the overall market. In order to make sure this program was successfully executed I needed to import 2 external libraries. First I downloaded the YFinance library. This library served its purpose by providing the program with all the financial information needed. It would give each stock's Price to Earnings Ratio and the Dividend Yield. It also offered the opening and closing price of the stock. The other library we needed was Matplotlib which would help us by graphing the stocks at the end. In the beginning the program would greet its users by supplying them with some directions.

```
Welcome to your personal Investment Advisory Program!
Directions are as follows:
1.) Enter the tickers for the 2 stocks you would like to compare
2.) The program will use the Price to Earnings ratio and the Dividend Yield to calculate which is the better investment
3.) The program will then graph the stock it chose with the S and P 500
4.) If this program is unable to indentify one stock due to these two metrics you will be notified. If this occurs both stocks will be graphed
Disclaimer: In order for this program to work it is important that you pick stocks which offer both a P/E value and a Dividend Yield
```

Next the user would be asked to enter the tickers of the 2 stocks it would like to compare. For the purpose of this report I decided to choose Microsoft(MSFT) and Apple(AAPL). After this is done the program initializes each stock ticker with the yfinance library using “yf.Ticker()”. Following this the program uses the yfinance library to extract the data it needs. This is done by using the info() option. The program on how this is done is shown below.

```
sandp500 = yf.Ticker("^GSPC")

divyld1 = company1.info["dividendRate"]
divyld2 = company2.info["dividendRate"]
pe1 = company1.info["trailingPE"]
pe2 = company2.info["trailingPE"]
```

We then used a if loop to then see if the first company had the higher dividend yield and lower P/E ratio. If not the program would move to the elif statement to see if the second company had the higher dividend yield and lower P/E ratio. If this also was not the case then the program would move to the else statement which would say that a conclusion was not reached and graph both stocks with the S and P 500. When we used MSFT and AAPL a conclusion was made and the output was as follows.

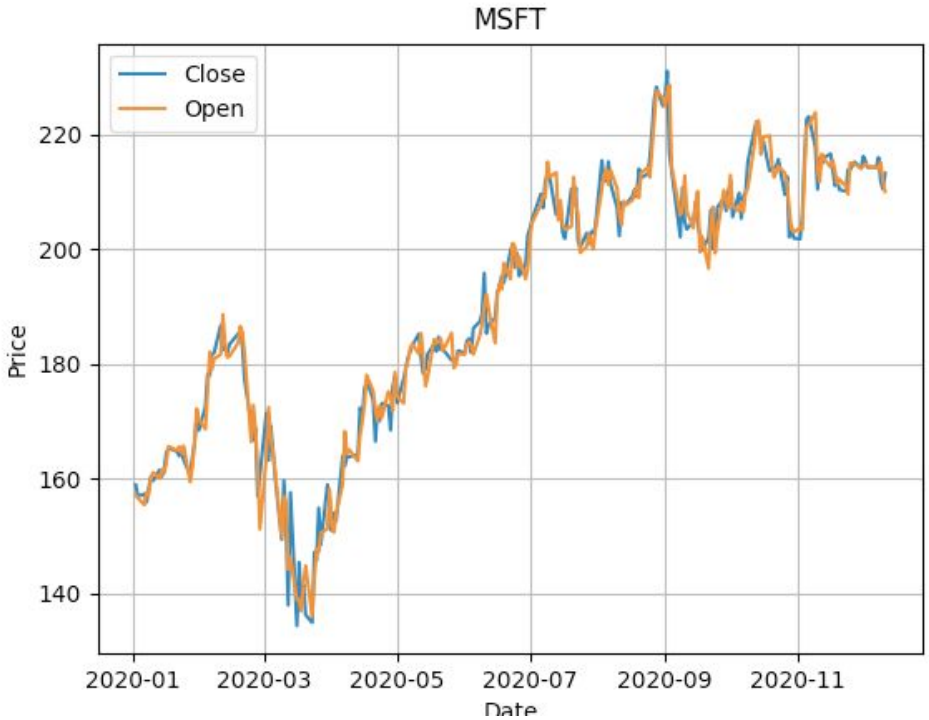
```
The two tickers you would like to compare are MSFT and AAPL
MSFT is the best investment if evaluated by the two metrics used
Here is the graph of the stock the program has identififed as the better investment graphed with the S and P 500
```

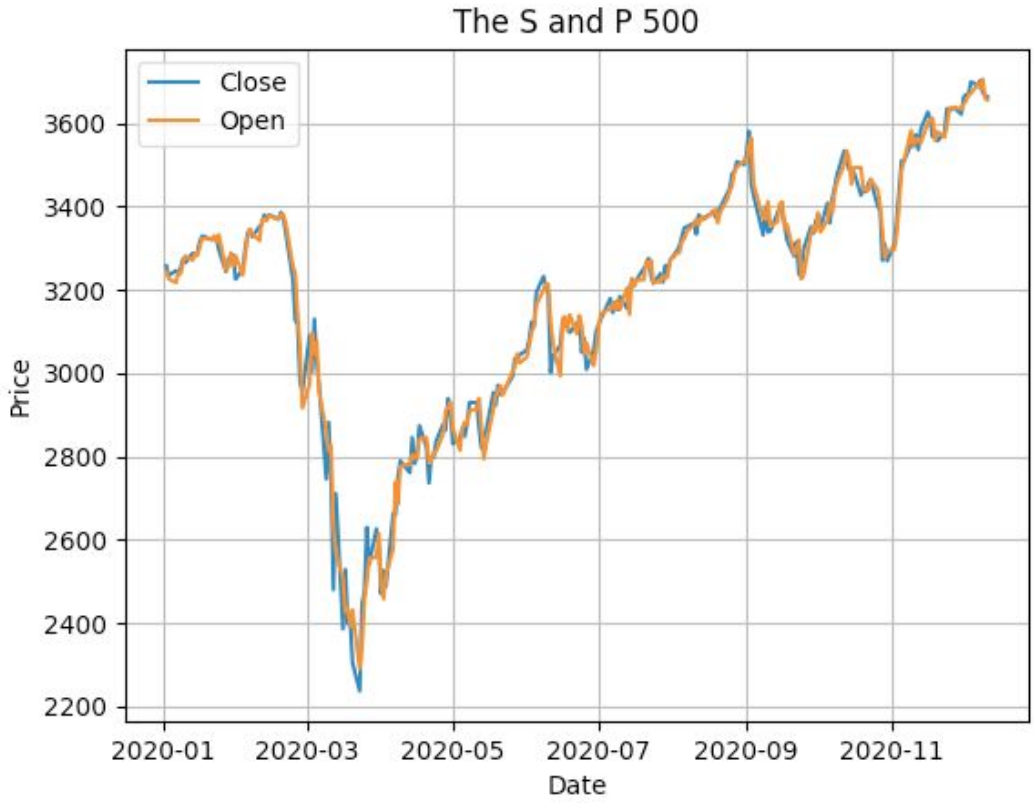
For the graphing parts I first would create a variable that would be filled by the history of that stock from 1/1/2020 to present. I would then have two different lines one with all the opening prices and one with the closing ones. I would then give the graph and each axes a title. I would follow it up by providing a key to show which is the opening and which is the closing price. The code for the graphing is below.

```
sandp = sandp500.history(start="2020-01-01")
plt.plot(sandp.index, sandp["Close"], label="Close")
plt.plot(sandp.index, sandp["Open"], label="Open")

plt.xlabel("Date")
plt.ylabel("Price")
plt.title("The S and P 500")
plt.legend(loc="upper left")
plt.grid()
plt.show()
```

In this particular example, MSFT was chosen to be the best investment. Here are a picture of each graph.





The program's last remarks were a thank you message to the user.