Connor Lucas

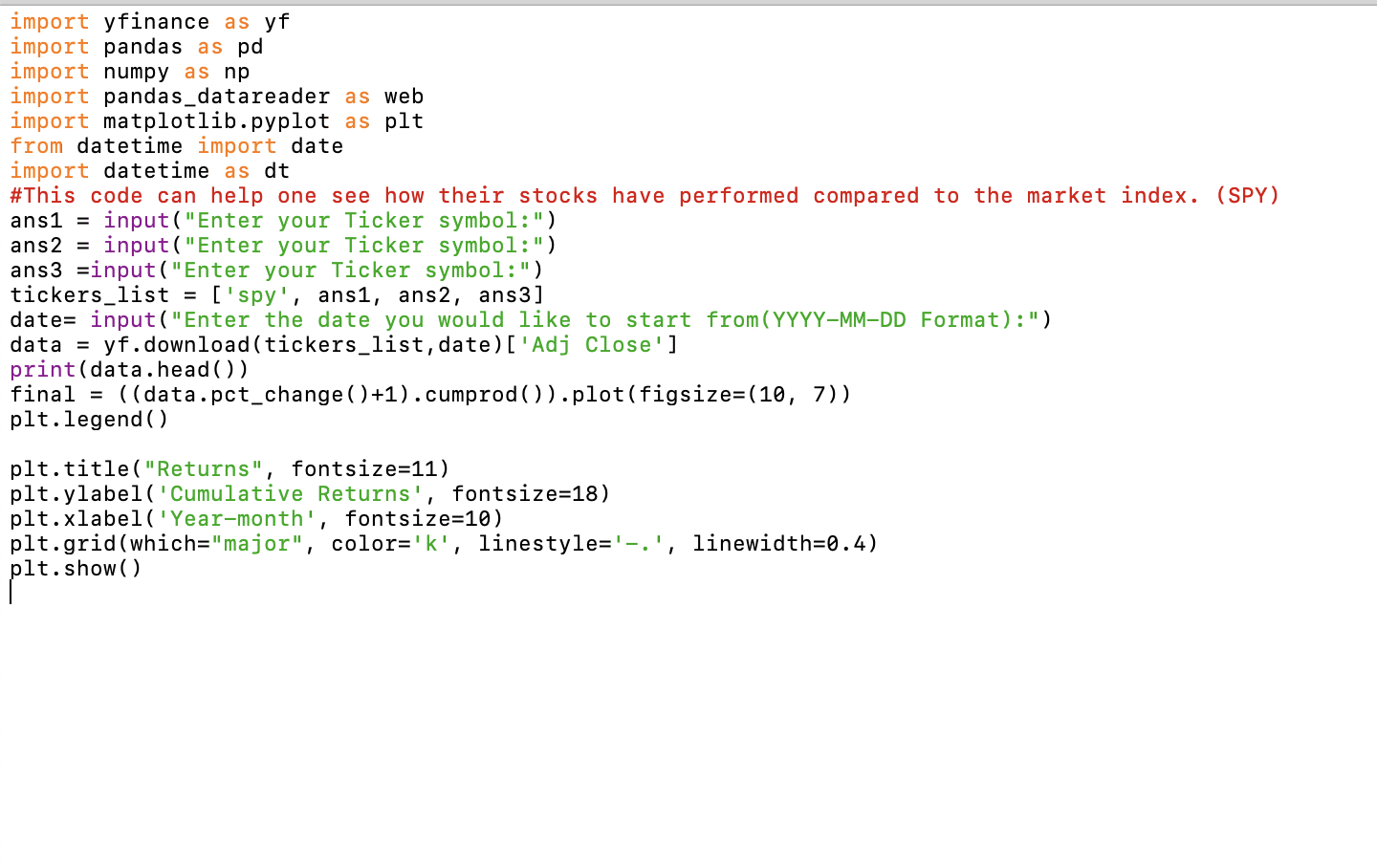
Professor Ryan

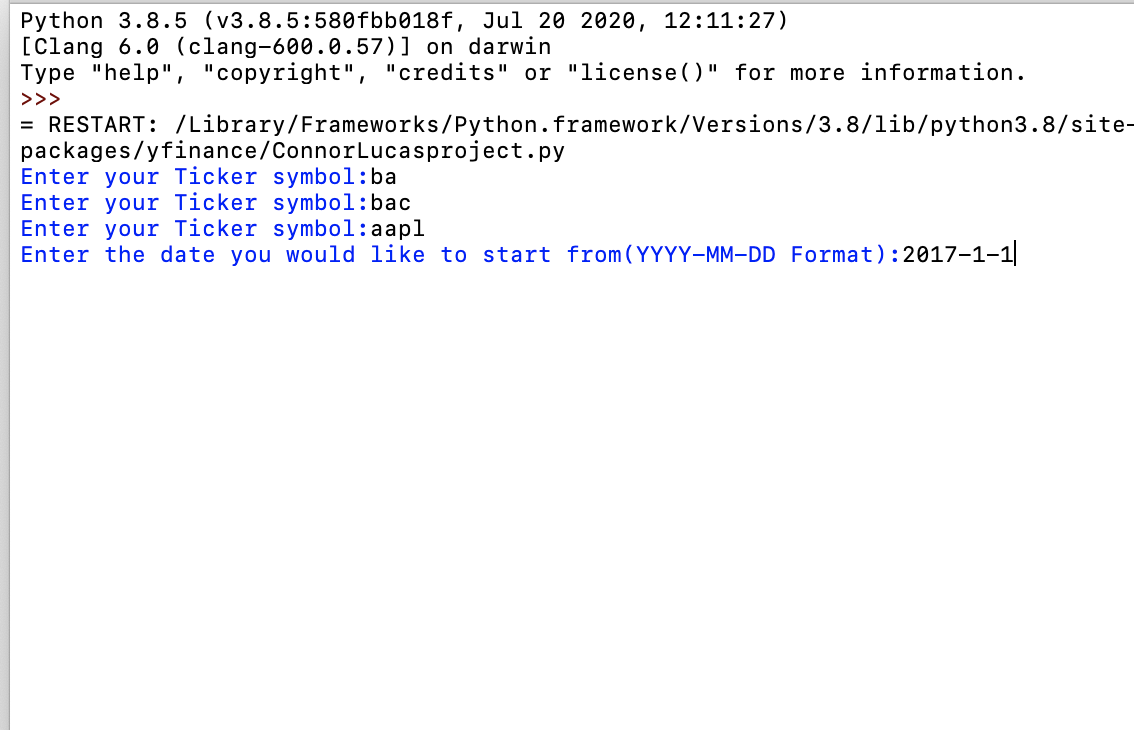
CS Final Project

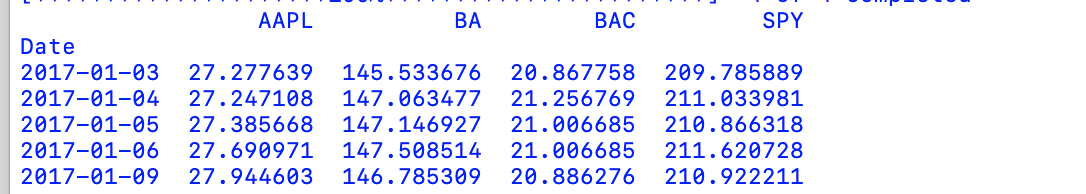
12 December 2020

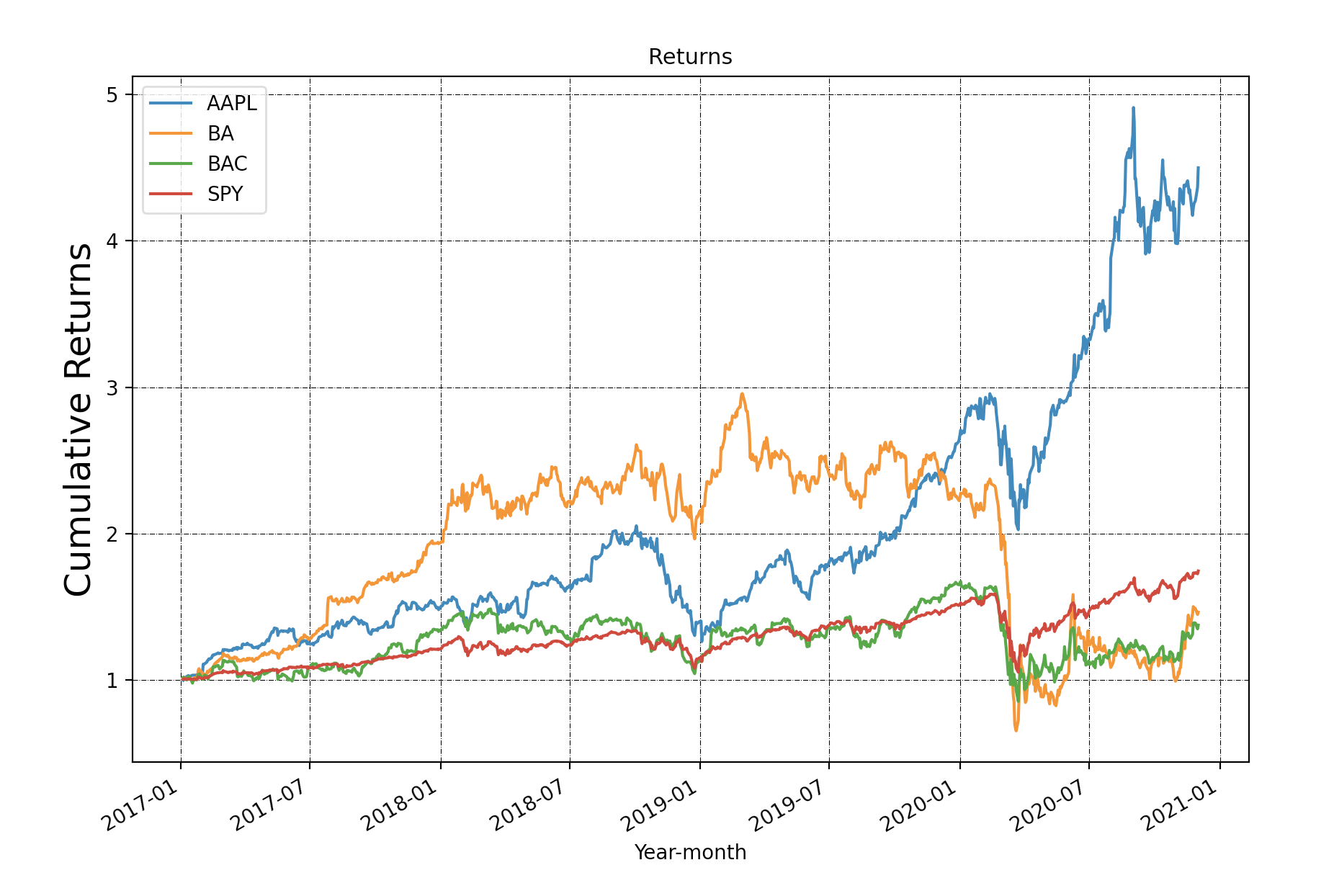
**Section 1:**

Many people wonder how a certain stock has done compared to the overall market during a certain time period. Many people start investing in individual stocks and want to see how they have done compared to the market index ticker symbol “SPY”. Unfortunately, many websites do not allow one the opportunity to choose a specific date to compare their certain ticker to the market. In my code, one is able to enter up to 3 stocks that are in their portfolio or that they are curious about and see if since the date they entered, (or would have entered) if they were able to outperform the market in that given time.



The input to this function is quite simple. First, it asks the user to name up to three stocks. Those three stocks are then combined with SPY into a list. That list is then taken and put into something called data where data from those stocks is compiled starting with whatever date the user entered. I have my data from yfinance and use matplotlib to plot my charts. This code is unique, because there are little to no other codes out there which let a user not only input any stock they want to be compared, but also allow the user to input any starting date they would like. This can be used by many trading platforms, as I am sure many people are curious to see how specific stocks have done compared to the market over any period of time. This gives a cumulative daily return to the user which is then plotted on a graph. I was able to make the graph, adn change the names of the legends and other labels to make it simple to understand. The colors are clearly distinguishable, giving the user a clear picture of their stocks and their returns. 





The output which shows the companies’ closes through each day from whatever day the user picked, so one can see if they could have bought at a better time. This is helpful because one is able to truly understand the true volatility of the market and see that although some closes may be worse than others, the market tends to overall increase. The output also produces a line graph so that the user can see their cumulative daily returns compared to the market and for their specific stock(s). This output can help one realize that with the correct purchases in companies such as AAPl and AMZN can return you amazing gains compared to the market over the last few years, but investments in companies such as XOM, can lead to one underperforming the market. This can help one identify a growth stock, or maybe even a value play, as they can see which stocks have underperformed the market.

**Section 2: Target Audience**

The target audience can be anyone curious about stocks, to a retail trader trying to find stocks with momentum, all the way to firms using this program to find out how their positions have done against the market in a quick and effective way. These firms can easily use this to show their clients their performance from any date against the market.

**Section 3: Specific Programming Techniques Used:**

I first import finance to get the data for stocks and also matplotlib, as this makes the graph for me and does some of the math for me. The program asks for two inputs, which the user puts in. It then puts your answer into a list, and runs all of it in a string until all of the stocks have been run through finance. The code then makes a plot by taking the percent change of the stock from every close, and using the cumprod() function to find the cumulative product of the percent change. The code gives a graphical representation of performance as well as listing every closing price of the stocks from the date the person imported. I also made a legend, which I will get into in the next section.

**Section 4: Hardest Part:**

The hardest part of this had to be making the legend in the graph and naming everything. This is because we had yet to learn it in class, as I started this project early. I messed up many times and fortunately was able to use Youtube in order to teach myself. Once professor Ryan taught the class, I then changed my code again to make it similar to what we had learned in class. If I would have waited a little longer to start, I would have learned it in class and been okay. I also had to look up the cumprod function in order to find cumulative daily returns.

**Section 5: What would I add?**

Given more time, I would have figured out how to ask the user what their initial investment was in each company at the time, and if any recurring investments occurred, and how frequently. With this information, I could have taken their investment and told them what their investment would be worth today. I am not sure if it is just me, but I always wonder if I invested this much in the market at this time, how much would I have. This would be a cool feature that I may definitely try and add.