Kayla Batzer

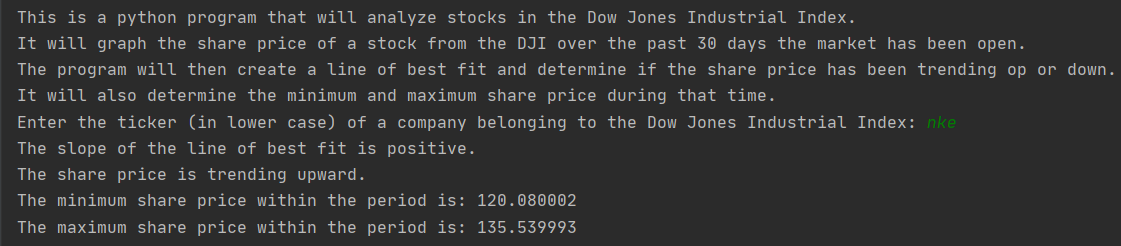
CS 110 Personal Project

Prof. Ryan

13 December 2020

**Section I: Overview and Summary**

The program that I created analyzes any stock from the Dow Jones Industrial Index. It starts by asking the user for a ticker input, then graphs the share price of that stock over the past thirty days ending on December first. Next, it graphs a line of best fit and determines the recent trend depending on whether the slope of the line is positive or negative. The program will then print the minimum and maximum share price over that time period.

Users of this program will first see a brief description of what the program does. They will then receive a prompt to enter the lowercase ticker of a stock in the Dow Jones Industrial index. Assuming that they enter a valid ticker symbol, they will then see the graph of the share price along with the line of best fit. They will be given the trend indicated by the line of best fit, as well as the maximum and minimum share price for the time period. For example, if the user were to enter the ticker “nke” for NIKE, the output would be as follows:



**Section II: Target Audience**

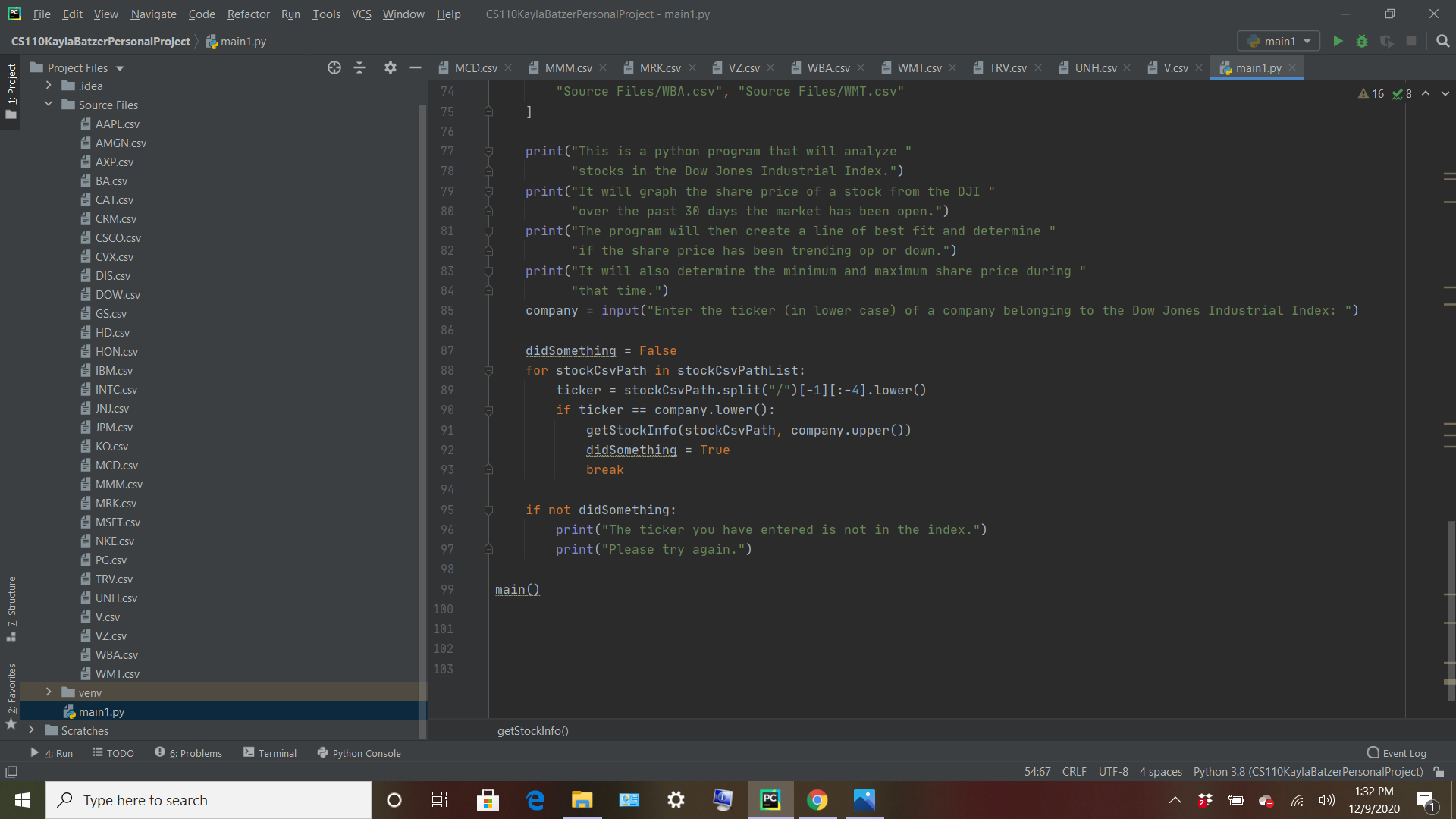
The target audience of this program could be anyone looking to gather information on recent trends of the stocks within the Dow Jones Industrial Index, however it is especially relevant to momentum investors. Momentum investors take advantage of current trends in the market to make their profit. This involves buying during uptrends and selling during downtrends. This program finds a line of best fit and determines the trend of the share price, which is pertinent information for that type of investing. While momentum investors typically look for trends in the moving average, this program, at the very least, provides supplementary information to consider while investing.

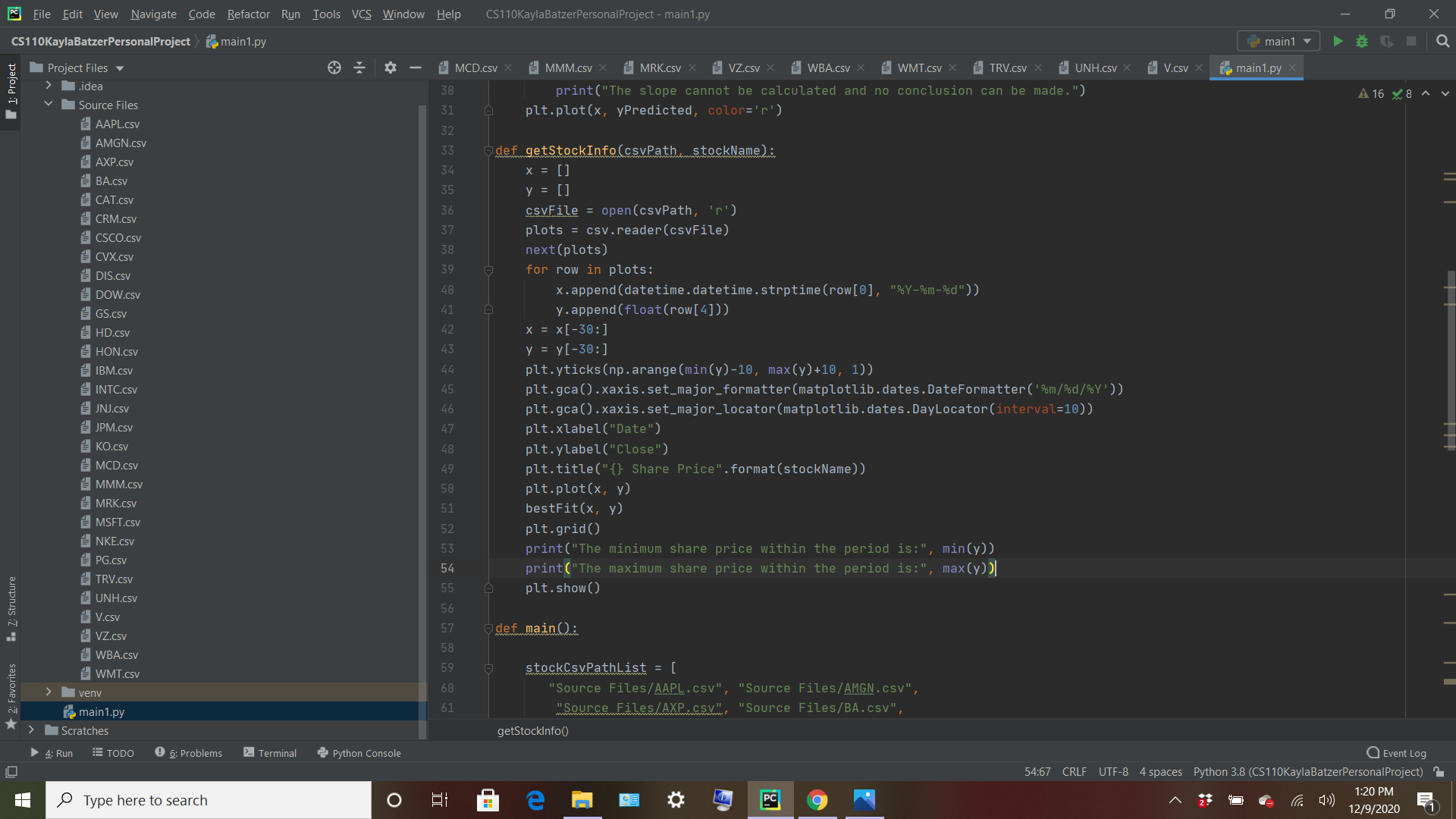
**Section III: Specific Programming Techniques**

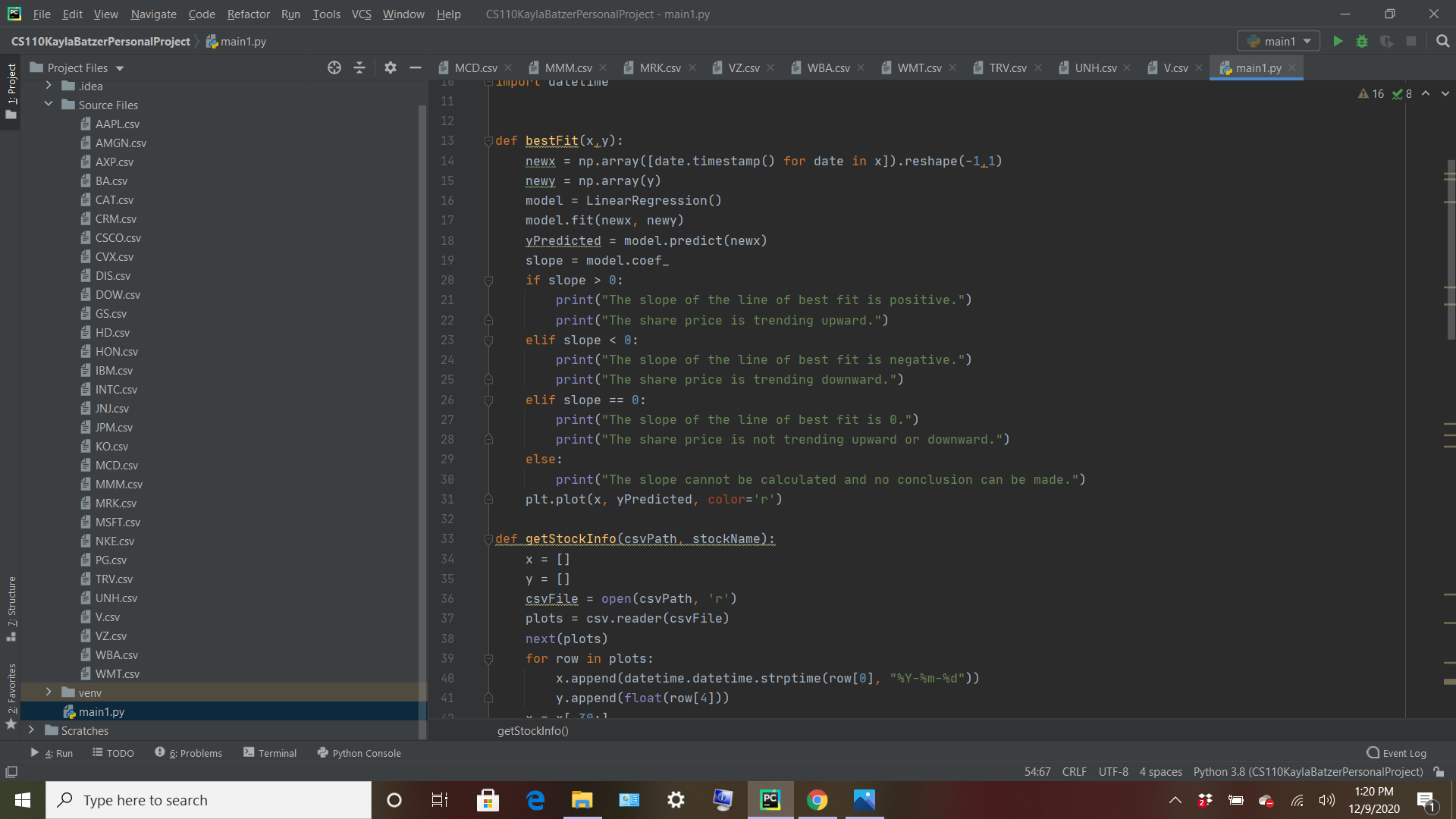
In order for this program to run, multiple different libraries had to be imported. The csv library was used to read the csv files containing the data and the matplotlib library was used to create the graph of the share price. The LinearRegression library was used to create the line of best fit and the numpy library was used to reshape the data in a way that would allow linear regression to be used. The matplotlib.dates and datetime libraries were used to configure the dates used on the x-axis of the graph. They had to be turned into datetime in order to be used in the linear regression model.

Decision structures were used twice throughout the program. Once was in the main to run the program if a valid ticker was entered or show an error message otherwise. The second time occurred within the function that graphs the line of best fit. If-elif-else statements were used to print the trend of the share price based on the slope of the line of best fit. Images of the decision structures can be found within the images of the functions.

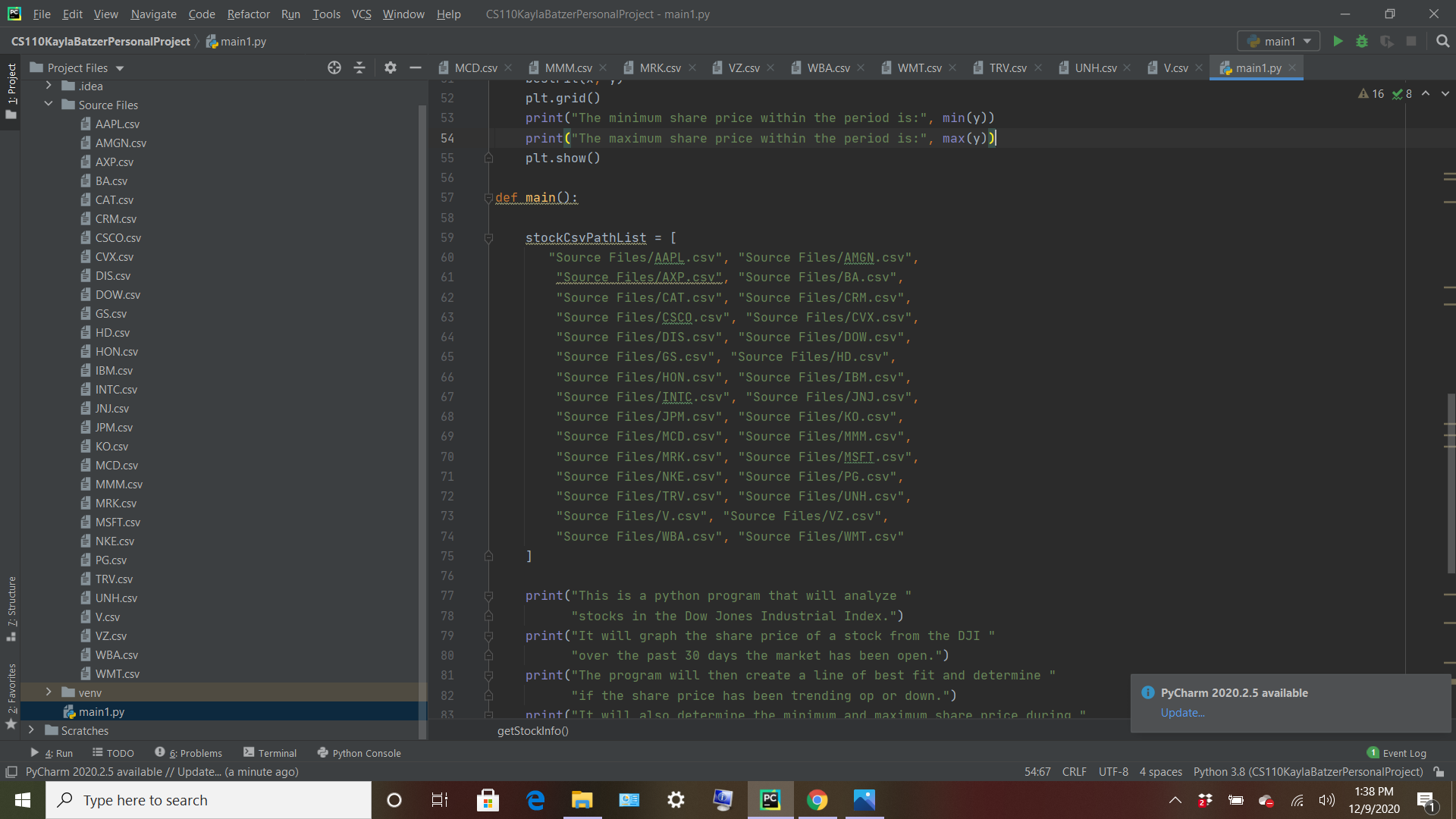
The program consists of three functions: “bestFit”, “getStockInfo”, and “Main”. The purpose of “bestFit” is to graph the line of best fit and determine the trend of the share price. “bestFit” is called within the function “getStockInfo”, which is a function that reads the csv file and creates the plot. “getStockInfo” also determines the minimum and maximum share price for the period. The “Main” function contains the list of csv files, asks the user for the input, and executes the program. Below are images of the functions (the main function excludes the csv file list):







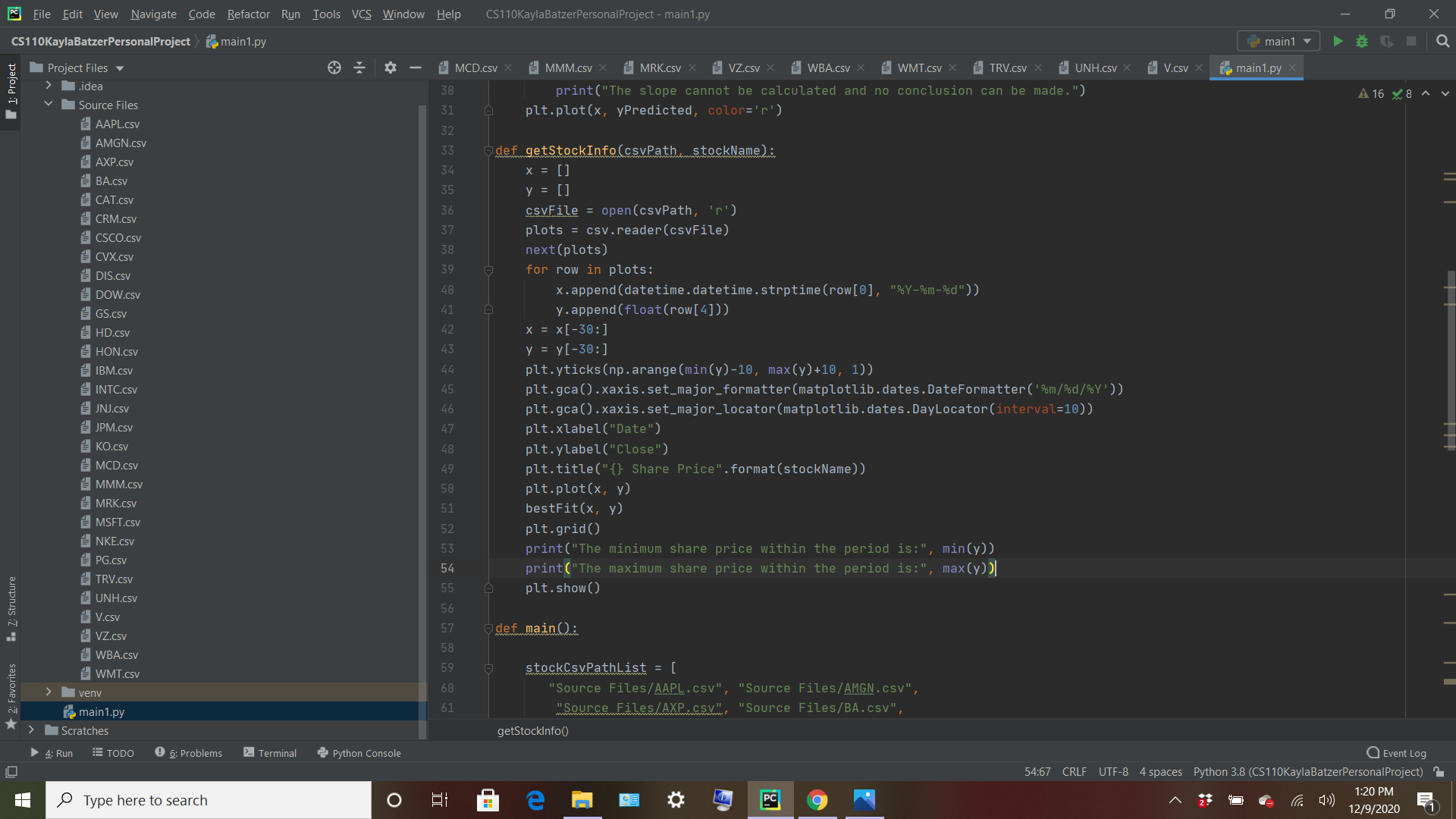
The data used in this program came from Yahoo Finance historical data. The historical data for each stock in the Dow Jones Industrial Index was downloaded from the website and used in the program. The csv files were placed in a folder within the project called “Source Files”, so that the data could easily be referred to and used within the program. The csv files were put into a list within the “Main” function of the program and the file used is dependent on the user’s input. Here is an image of that list:



**Section IV: Challenges**

The main issue that I encountered while designing this program was with the dates. Because the dates were strings, they could not be used in the regression model (it is impossible to take the slope of a line where the x coordinates are strings). With help I was able to convert the dates into datetime, which allowed the dates to be used in the linear regression while displaying the accurate dates on the x-axis.

Additionally, I had trouble with limiting the data from the csv files to the last thirty rows. Plotting all of the data gave me a graph that could not be read and made little sense, so I chose to limit it. Using plt.xlim() and plt.ylim() did not work no matter how I formatted it. Finally with help I was able to limit it by doing the following:



**Section V: Future Extensions**

One way to improve this program would be to make it so the user can determine the time period that the stock is analyzed in. Currently, the program analyzes a set thirty days, so it could be beneficial to widen the program’s reach by expanding the time period.

Another way to improve the program would be to expand which stocks can be analyzed. Right now, the program is limited to stocks within the Dow Jones Industrial Index. Using the yfinance library, the user could analyze any stock, not just those within the DJI.

Lastly, the inner workings of the program can be improved by graphing the moving average of the stock, rather than just the share price. The moving average is more useful to momentum investors (the program’s target audience), so implementing this would improve the practicality of the program.