**Project Overview**

The performance of “Green Stock” portfolio was analyzed over 2 years to determine its viability for investment. To perform this analysis, I utilized a Visual Basic Application tool in an Excel Workbook, this would help to determine the stock’s daily volumes and annual return over the above specified time frame. A combination of stocks was analyzed to determine what was the best option for investing. Subsequently, I then compared the effectiveness of the same analysis in VBA on a larger data set.

**Purpose**

The purpose of this analysis was to determine the most expedient and efficient way to observe the performance of multiple stocks in VBA. After completing an analysis of the initial stocks’ portfolio, it was determined that it was very efficient to analyze the stocks based on the data given. To obtain a more efficient analysis, I then refactored the code to determine if a better and faster analysis of the stocks could be obtained for analysis if the data set expanded to a larger set.

**Process**

**Refactoring the Code**

I refactored the code to make it more efficient by changing my loops and utilizing 4 different arrays: tickers, tickerVolumes, tickerStartingPrices and tickerEndingPrices. These labels/variables were used to determine the tickers of the stocks. I also created a tickerIndex variable to match the other labels. By doing this I was able to assign values to each ticker before iterating in the data set which allowed the analysis process to be more efficient.

**Results**

The stock returns performed better in 2017 yielding more GREEN outputs versus RED. In 2018, stock returns yielded majority negative returns for our chosen tickers.

The 2017 refactored code ran in approximately .115 seconds. The 2018 refactored code ran in approximately .107 seconds. Both years ran very quick and runtimes for each year were very close for the provided datasets.

**Advantages of refactoring code**

The code was refactored to ensure that it ran speedily when analyzing a large data set of stock tickers.

Some of the **advantages** were:

* Less time to run the code: This allows us to run our code more often if needed, such as if we want to run more than 2 years. This could be done much more efficiently with our refactored code.
* Able to run larger data set: We are now able to run more tickers on top of running more years. Refactored code allows us to be able to do this expeditiously.

While one of the **disadvantages** was:

* Slightly more complex code: This can lead to errors in compiling or in output due to increasing the chances of coding errors as the code becomes more complex.