**Overview**

The project is to create a VBA script in Excel to analyze stock data and calculate the overall worth of each stock and the projected return investment for selected years.

**Results**

The results of the refactored code are substantial. The code ran at a fraction of the time of the original. As depicted with the below snippets:

Graphical user interface, application, table, Excel

Description automatically generated

As you can see, the time it took to run the script is 0.8 seconds. However with the new script the results were given much faster as depicted below:

Graphical user interface, application, table, Excel

Description automatically generated

With the refactored script, the code ran at a seventh of the original time. This also occurred for the year 2018 with speed results being similar.

**Summary**

Refactoring code is commonplace in many tech fields. Refactoring is using someone else’s code and modifying it yourself for your own needs. It’s like taking a template and adding in the words. It’s advantageous as it can save time as you won’t need to write the entire code yourself by refactoring. Refactoring can shorten the overall time the code needs to run as well, shown in my results above. While refactoring can be helpful, it comes with disadvantages. Using someone else’s code means you must know what their code is doing and the purpose you need it for. If you are unfamiliar with the code you’re refactoring, you could potentially risk the security of your computer as some code can be extremely malicious. If you understand what your code is doing, refactoring is useful and efficient.