VBA Challenge

Overview of the Project

For the Module 2 Challenge, we had to take the workbook that we completed for Steve and refactor the original code to make it run more efficiently so that it can handle a larger amount of data at one time. The original code was used to run an analysis for a few select stocks in the Stock Market to see which stocks would be the best choice for Steve’s parents to invest in.

Results

The 2017 stock market performed a lot better than the 2018 stocks which was made clear after refactoring the original code because the execution time for the edited code was reduced greatly. Below is my code that we were asked to refactor, the runtimes of the 2017 and 2018 VBA Macros and their performance data:

'1) Format the output sheet on All Stocks Analysis worksheet

'Create a header row

Worksheets("All Stocks Analysis").Activate

Range("A1").Value = "All Stocks (2018)"

Cells(3, 1).Value = "Ticker"

Cells(3, 2).Value = "Total Daily Volume"

Cells(3, 3).Value = "Return"

'2) Initialize array of all tickers

Dim tickers(12) As String

tickers(0) = "AY"

tickers(1) = "CSIQ"

tickers(2) = "DQ"

tickers(3) = "ENPH"

tickers(4) = "FSLR"

tickers(5) = "HASI"

tickers(6) = "JKS"

tickers(7) = "RUN"

tickers(8) = "SEDG"

tickers(9) = "SPWR"

tickers(10) = "TERP"

tickers(11) = "VSLR"

'3a) Initialize variables for starting price and ending price

Dim startingPrice As Single

Dim endingPrice As Single

'3b) Activate data worksheet

Worksheets("2018").Activate

'3c) Get the number of rows to loop over

RowCount = Cells(Rows.Count, "A").End(xlUp).Row

'4) Loop through tickers

For i = 0 To 11

ticker = tickers(i)

totalVolume = 0

'5) loop through rows in the data

Worksheets(yearValue).Activate

For j = 2 To RowCount

'5a) Get total volume for current ticker

If Cells(j, 1).Value = ticker Then

totalVolume = totalVolume + Cells(j, 8).Value

End If

'5b) get starting price for current ticker

If Cells(j - 1, 1).Value <> ticker And Cells(j, 1).Value = ticker Then

startingPrice = Cells(j, 6).Value

End If

'5c) get ending price for current ticker

If Cells(j + 1, 1).Value <> ticker And Cells(j, 1).Value = ticker Then

endingPrice = Cells(j, 6).Value

End If

Next i

'6) Output data for current ticker

Worksheets("All Stocks Analysis").Activate

Cells(4 + i, 1).Value = ticker

Cells(4 + i, 2).Value = totalVolume

Cells(4 + i, 3).Value = endingPrice / startingPrice - 1

Next i

Graphical user interface, application

Description automatically generated Graphical user interface, application, table, Excel

Description automatically generated

Graphical user interface, application

Description automatically generated Table

Description automatically generated

Summary

Advantages of Refactoring Code

The main advantage that I saw for refactoring code was that the end product showed up faster because the code script, itself was more straightforward and concise. What I liked about the refactored code was that it looked cleaner and easier to read at the end, which would help out anyone who is currently working with the code or looking at the code sometime in the future.

As you can see from my screenshots above, the macro runtime is fast, however with the original code before the refactoring, it most definitely did not run that efficiently. The only issue that I could think of when refactoring the original code was that it was time-consuming.