Sub AllStocksAnalysisRefactored\_ChallengeAssignment()

Dim startTime As Single

Dim endTime As Single

yearValue = InputBox("What year would you like to run the analysis on?")

startTime = Timer

'Format the output sheet on All Stocks Analysis worksheet

Worksheets("All Stocks Analysis").Activate

Range("B1").Value = "All Stocks (" + yearValue + ")"

Range("B1").Font.FontStyle = "Bold"

'Create a header row

Cells(3, 2).Value = "Ticker"

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

Cells(3, 4).Value = "Return"

Range("B3:D3").Interior.ColorIndex = 15

'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"

'Activate data worksheet

Worksheets(yearValue).Activate

'Get the number of rows to loop over

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

'DELIVERABLE - REQUIREMENTS

'1a The tickerIndex is set equal to zero before looping over the rows.

For i = 0 To 11

tickerIndex = tickers(i)

'Arrays are created for tickers, tickerVolumes, tickerStartingPrices, and tickerEndingPrices

Dim tickerVolumes As Long

Dim tickerStartingPrices As Single, tickerEndingPrices As Single

'The tickerIndex is used to access the stock ticker index for the tickers, tickerVolumes, tickerStartingPrices, and tickerEndingPrices arrays

Worksheets(yearValue).Activate

tickerVolumes = 0

For j = 2 To RowCount

'If the next row’s ticker doesn’t match, increase the tickerIndex.

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

'Increase volume for current ticker

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

End If

'The script loops through stock data, reading and storing all of the following values from each row:

'tickers, tickerVolumes, tickerStartingPrices, and tickerEndingPrices

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

'Store Starting Price Value

tickerStartingPrices = Cells(j, 6).Value

End If

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

'Store Ending Price Value

tickerEndingPrices = Cells(j, 6).Value

End If

Next j

'Code for formatting the cells in the spreadsheet is working.

Worksheets("All Stocks Analysis").Activate

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

Cells(4 + i, 3).Value = tickerVolumes

Cells(4 + i, 4).Value = tickerEndingPrices / tickerStartingPrices - 1

'Fix % on return

With Range("D4:D15")

.NumberFormat = "0.0%"

.Value = .Value

End With

Next i

'Formatting

Worksheets("All Stocks Analysis").Activate

Range("B3:D3").Font.FontStyle = "Bold"

Range("B3:D3").Borders(xlEdgeBottom).LineStyle = xlContinuous

Range("C4:C15").NumberFormat = "#,##0"

'Range("C4:C15").NumberFormat = "0.0%"

Columns("C").AutoFit

dataRowStart = 4

dataRowEnd = 15

For i = dataRowStart To dataRowEnd

If Cells(i, 4) > 0 Then

Cells(i, 4).Interior.Color = vbGreen

Else

Cells(i, 4).Interior.Color = vbRed

End If

Next i

endTime = Timer

MsgBox "This code ran in " & (endTime - startTime) & " seconds for the year " & (yearValue)

End Sub