Sub ModChallengeVBAAnalysis()

' Loop through all worksheets in the workbook

For Each ws In ThisWorkbook.Worksheets

ws.Activate

' Define variables

Dim Opened As Double

Dim Closed As Double

Dim YearlyChange As Double

Dim PercentChange As Double

Dim MaxIncrease As Double

Dim MaxDecrease As Double

Dim MaxVolume As Double

Dim Ticker As String

Dim LastRow As Long

Dim LastTickerRow As Long

Dim SummaryRow As Long

Dim TotalVolume As Double

' Initialize variables (declare)

Opened = Cells(2, 3).Value

TotalVolume = 0

SummaryRow = 2

' Find the last row with data in column A

LastRow = Cells(Rows.Count, 1).End(xlUp).Row

' Loop for all the rows

For i = 2 To LastRow

Ticker = Cells(i, 1).Value

TotalVolume = TotalVolume + Cells(i, 7).Value

' Check the ticker to see if we're on a different stock

If Ticker <> Cells(i + 1, 1).Value Then

Closed = Cells(i, 6).Value

' Solving for YearlyChange and PercentChange

YearlyChange = Closed - Opened

If Opened <> 0 Then

PercentChange = (Closed - Opened) / Opened

Else

PercentChange = 0

End If

' Output the results in the summary table

ws.Cells(SummaryRow, 9).Value = Ticker

ws.Cells(SummaryRow, 10).Value = YearlyChange

ws.Cells(SummaryRow, 11).Value = PercentChange

ws.Cells(SummaryRow, 12).Value = TotalVolume

' Conditional formatting

If YearlyChange < 0 Then

ws.Cells(SummaryRow, 10).Interior.Color = RGB(255, 0, 0) ' Red

ElseIf YearlyChange > 0 Then

ws.Cells(SummaryRow, 10).Interior.Color = RGB(0, 255, 0) ' Green

End If

' Update SummaryRow and reset variables for the next ticker

SummaryRow = SummaryRow + 1

Opened = Cells(i + 1, 3).Value

TotalVolume = 0

End If

Next i

' Search for the last row in the summary table

LastTickerRow = Cells(Rows.Count, 9).End(xlUp).Row

' Find the stock with the Greatest % Increase, Greatest % Decrease, and Greatest Total Volume

For j = 2 To LastTickerRow

If Cells(j, 11).Value > MaxIncrease Then

MaxIncrease = Cells(j, 11).Value

ws.Cells(2, 15).Value = Cells(j, 9).Value

ws.Cells(2, 16).Value = MaxIncrease

End If

If Cells(j, 11).Value < MaxDecrease Then

MaxDecrease = Cells(j, 11).Value

ws.Cells(3, 15).Value = Cells(j, 9).Value

ws.Cells(3, 16).Value = MaxDecrease

End If

If Cells(j, 12).Value > MaxVolume Then

MaxVolume = Cells(j, 12).Value

ws.Cells(4, 15).Value = Cells(j, 9).Value

ws.Cells(4, 16).Value = MaxVolume

End If

Next j

Next ws

End Sub