Sub stock\_analysis()

'Set the variables

'ticker name

Dim TickerSymbol As String

'Set an initial variable for holding the total volume per ticker

Dim Ticker\_Volume As Double

'Keep track of the location for each Ticker in the summary table

Dim Ticker\_Table\_Row As Integer

'Following variables set the opening price (year beginning), closing price (year end), boolean for tracking, Annual price change, percent change, arrays to track Opening & Closing prices, counter to loop through array

'Challenge section variables for maximum volume, maximum decrease, maximum increase pcts

Dim openPrice As Double

Dim closePrice As Double

Dim openBool As Boolean

Dim yearlyChange As Double

Dim pctChange As Double

Dim openPriceArray() As Double

Dim closePriceArray() As Double

Dim arrayCounter As Integer

Dim maxVolume As Double

Dim maxPctChange As Double

Dim minPctChange As Double

Dim summaryRow As Integer

'Begin the worksheet loop

Dim ws\_num As Integer 'no. of worksheets

Dim ws\_counter As Integer 'counter to track worksheets

Dim shtName As String 'active worksheet name

Dim starting\_ws As Worksheet

Set starting\_ws = ActiveSheet 'remember which worksheet is active in the beginning

ws\_num = ThisWorkbook.Worksheets.Count 'get # of worksheets

'counters to track rows (tickers) and arrays

Dim i As Long

Dim k As Integer

'Start the loop for each Worksheet:

For ws\_counter = 1 To ws\_num

ThisWorkbook.Worksheets(ws\_counter).Activate

shtName = ActiveWorkbook.Worksheets(ws\_counter).Name

Ticker\_Table\_Row = 2

'Determine last row in ticker column:

Dim LastRow As Long

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

'Set the Summary table headers

Range("I1").Value = "Ticker"

Range("J1").Value = "Yearly Change"

Range("K1").Value = "Percent Change"

Range("L1").Value = "Total Stock Volume"

'To debug

'Range("N1").Value = "Closing Price"

'Range("O1").Value = "Opening Price"

Range("A1:O1").Font.Bold = True

'set the ticker volume sum

Ticker\_Volume = 0

openBool = False 'Set counter as False

yearlyChange = 0 'set initial values

pctChange = 0 'set initial values

arrayCounter = 0 'set initial values

'Copy unique values of tickers to a Z1 column - this is to redim the arrays

Dim tCnt As Integer

Range("A1:A" & Cells(Rows.Count, "A").End(xlUp).Row).AdvancedFilter \_

Action:=xlFilterCopy, CopyToRange:=Range("Z1"), Unique:=True

tCnt = Cells(Rows.Count, "Z").End(xlUp).Row

'Now provide the size for the array based on actual ticker counts

ReDim openPriceArray(tCnt)

ReDim closePriceArray(tCnt)

'clear the column:

Worksheets(shtName).Columns("Z").ClearContents

'Now loop each row in this active WS

For i = 2 To LastRow

' Check if we are still within the same Ticker, if it is not...

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

' Set the Ticker

TickerSymbol = Cells(i, 1).Value

' Add to the Volume Total

Ticker\_Volume = Ticker\_Volume + Cells(i, 7).Value

' Print the Ticker in the Summary Table

Range("I" & Ticker\_Table\_Row).Value = TickerSymbol

' Print the Brand Amount to the Summary Table

Range("L" & Ticker\_Table\_Row).Value = Ticker\_Volume

' Closing price

closePrice = Cells(i, 6).Value

'Range("N" & Ticker\_Table\_Row).Value = closePrice

closePriceArray(arrayCounter) = closePrice

' Add one to the summary table row

Ticker\_Table\_Row = Ticker\_Table\_Row + 1

'increment arrayCounter

arrayCounter = arrayCounter + 1

'Reset the Brand Total for next ticker

Ticker\_Volume = 0

'Reset openBool

openBool = False

' If the cell immediately following a row is the same Ticker...

Else

' Add to the Ticker Volume

Ticker\_Volume = Ticker\_Volume + Cells(i, 7).Value

' get the opening price

If openBool = False Then

openPrice = Cells(i, 3).Value

'Range("O" & Ticker\_Table\_Row).Value = openPrice

openPriceArray(arrayCounter) = openPrice

openBool = True

End If

End If

'End the loop through rows

Next i

'Next section prints the open & close changes, percentages

For k = LBound(openPriceArray) To (Ticker\_Table\_Row - 3)

'Year end price minus the starting price at the year beginning

yearlyChange = closePriceArray(k) - openPriceArray(k)

'Set the value of the annual change

Range("J" & (k + 2)).Value = yearlyChange

'Calculate the percentage & format to 2 decimals

'Account for a zero:

If openPriceArray(k) = 0 Then

pctChange = 0

Range("K" & (k + 2)).Value = Format(pctChange, "Standard")

Else

pctChange = (yearlyChange / openPriceArray(k)) \* 100

Range("K" & (k + 2)).Value = Format(pctChange, "Standard")

End If

'Color the rows: red if price decreased & green if price increased

If yearlyChange < 0 Then

Range("J" & (k + 2)).Interior.ColorIndex = 3

Else

Range("J" & (k + 2)).Interior.ColorIndex = 4

End If

Next k

'Challenge section

'Set the titles

Range("R1").Value = "Summary"

Range("S1").Value = "Ticker"

Range("T1").Value = "Summary Value"

Range("R2").Value = "Greatest % Increase"

Range("R3").Value = "Greatest % Decrease"

Range("R4").Value = "Greatest Stock Volume"

Range("R1:T1").Font.Bold = True

'Find the max Volume

maxVolume = Application.WorksheetFunction.Max(Columns("L"))

Range("T4").Value = maxVolume 'set the max volume

summaryRow = Worksheets(shtName).Columns("L").Find(maxVolume).Row 'look for ticker that matches the max ticker volume

Range("S4").Value = Range("I" & summaryRow)

'Find the greatest % increase

maxPctChange = Application.WorksheetFunction.Max(Columns("K")) 'Max pct change

Range("T2").Value = maxPctChange

summaryRow = Worksheets(shtName).Columns("K").Find(maxPctChange).Row 'look for ticker that matches the max pct change

Range("S2").Value = Range("I" & summaryRow)

'Find the greatest % decrease

minPctChange = Application.WorksheetFunction.Min(Columns("K")) 'min pct change

Range("T3").Value = minPctChange

summaryRow = Worksheets(shtName).Columns("K").Find(minPctChange).Row 'look for ticker that matches the min pct change

Range("S3").Value = Range("I" & summaryRow)

'autofit the colums

Worksheets(shtName).Columns("A:Z").AutoFit

ActiveWindow.Zoom = 80

Next ws\_counter

End Sub