VBA Script for Challenge 1 (Greatest)

```
Sub Stock_Analysis_Greatest()

' Define the variables that will be used

        Dim Ticker As String
        Dim Volume_Total As Double
            Volume_Total = 0
        Dim Open_Amt As Double
            Open_Amt = Cells(2, 3).Value
        Dim Close_Amt As Double
            Close_Amt = 0
        Dim Yearly_Change As Double
            Yearly_Change = 0
        Dim Percent_Change As Double
            Percent_Change = 0

        Dim Greatest_Perc_Inc_Ticker As String
        Dim Greatest_Perc_Dec_Ticker As String
        Dim Greatest_Total_Volume_Ticker As String
        Dim Greatest_Percent_Increase As Double
            Greatest_Percent_Increase = 0
        Dim Greatest_Percent_Decrease As Double
            Greatest_Percent_Decrease = 0
        Dim Greatest_Total_Volume As Double
            Greatest_Total_Volume = 0

        Dim Summary_Table_Row As Long
            Summary_Table_Row = 2

        Dim Lastrow As Long
            Lastrow = Cells(Rows.Count, 1).End(xlUp).Row

' Label headers for the summary tables

        Range("K1,R1") = "Ticker"
        Range("L1") = "Yearly Change"
        Range("M1") = "Percent Change"
        Range("N1") = "Total Stock Volume"
        Range("S1") = "Value"
        Range("Q2") = "Greatest % Increase"
        Range("Q3") = "Greatest % Decrease"
        Range("Q4") = "Greatest Total Volume"
```

```
' Create loop to go through all the Tickers to find the required info

        For i = 2 To Lastrow

' Create if-then statement to check and see if we are still in the same Ticker
and set parameters based on that

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

' If the Tickers are not equal, then the following values will be true

                Ticker = Cells(i, 1).Value
                Close_Amt = Cells(i, 6).Value
                Yearly_Change = Close_Amt - Open_Amt

' Print variables into the appropriate cells

                Range("K" & Summary_Table_Row).Value = Ticker
                Range("L" & Summary_Table_Row).Value = Yearly_Change

' If Yearly Change is greater than zero, set the color to green

                    If Yearly_Change > "0" Then
                        Range("L" & Summary_Table_Row).Interior.ColorIndex = 4

' If Yearly Change is less than or equal to zero, set the color to red

                    ElseIf Yearly_Change <= "0" Then
                        Range("L" & Summary_Table_Row).Interior.ColorIndex = 3

                    End If

' Create another if-then statement to get the Percent Change and make sure to
account for the zero division possible error

                    If Open_Amt <> 0 Then
                        Percent_Change = (Yearly_Change / Open_Amt) * 100

' Print variables into cells and format the Percent Change to show numbers as a
percent

                        Range("M" & Summary_Table_Row).Value = Percent_Change
                        Range("M" & Summary_Table_Row).NumberFormat = "0.00\%"
```

```
' If the open amount is equal to zero, then Percent Change will be zero because
you cannot divide by zero

                    Else
                        Percent_Change = 0

                    End If

' Find the Greatest Percent Change

                    If Percent_Change > Greatest_Percent_Increase Then

' If the Percent Change is greater than the Greatest Percent Increase, then the
following is true:

                        Greatest_Percent_Increase = Percent_Change
                        Greatest_Perc_Inc_Ticker = Ticker

' Print Ticker with the Greatest Percent Increase and the number into the
appropriate cells with correct formatting

                        Range("R2").Value = Ticker
                        Range("S2").Value = Greatest_Percent_Increase
                        Range("S2").NumberFormat = "0.00\%"

' Find the Greatest Percent Decrease

                    ElseIf Percent_Change < Greatest_Percent_Decrease Then

' If the Percent Change is less than the Greatest Percent Decrease, then the
following is true:

                        Greatest_Percent_Decrease = Percent_Change
                        Greatest_Perc_Dec_Ticker = Ticker

' Print Ticker with the Greatest Percent Decrease and the corresponding percent
into the appropriate cells with correct formatting

                        Range("R3").Value = Ticker
                        Range("S3").Value = Greatest_Percent_Decrease
                        Range("S3").NumberFormat = "0.00\%"

                    End If
```

```
' Obtain Volume Total by Ticker name since we need to account for the last volume
amount before Ticker changes to next Ticker

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

' Print the Volume Total into the appropriate cells

                Range("N" & Summary_Table_Row).Value = Volume_Total

' Find the Greatest Total Volume

                    If Volume_Total > Greatest_Total_Volume Then

' If the Volume Total amount is greater than the Greatest Total Volume, then the
following is true:

                        Greatest_Total_Volume = Volume_Total
                        Greatest_Total_Volume_Ticker = Ticker

' Print Ticker with the Greatest Total Volume and the number into the appropriate
cells

                        Range("R4").Value = Ticker
                        Range("S4").Value = Greatest_Total_Volume

                    End If

' Add 1 to the Summary Table Row so that the correct variable is in the correct
cell for the next iteration

                Summary_Table_Row = Summary_Table_Row + 1

' Re-set to zero for next iteration

                Yearly_Change = 0
                Percent_Change = 0
                Close_Amt = 0
                Volume_Total = 0

' Get Open Amount for next Ticker

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

' If the Ticker value is equal, then the Volume Total would be 0 plus the Volume
Total numbers associated with the first Ticker
```

```
            Else

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

' Need to end this statement

            End If

' Need to end the loop

        Next i

' Format cells to autofit in the worksheet so that data is readable and looks
clean

      Cells.EntireColumn.AutoFit

' Must end the function

End Sub
```