NOTE: I wasn’t sure what to submit, there were references to screenshots, etc, rather than the Excel workbook with code, so this Word doc contains my sub to do the homework. I also submitted Excel workbook, screenshot of the completed worksheet. My code is heavily commented, as I worked through the various bits, hope it helps.

Sub Stock\_Summary()

'Carl Colburn

'UCI Data Analysis Bootcamp, Homework Assignment, Module 2, VBA

'NOTE some of my labels are legacy VB6, like identifying data type in var names ("str")

'dim variables

Dim i, j As Integer 'i for rows of dataset, j for rows of summary table

Dim x, y As Double 'can't be int due to potential high row number

Dim k As Double

'create variables to hold data as iterate through data sets

Dim strTicker As String

Dim dblYearlyChange, dblPercentChange, dblTotalVolume As Double 'holds numerics

Dim dblMin, dblMax, dblHighest As Double 'these will be for finding most pos or neg, highest vol

'need variables to hold min, max, highest. See top of sub

Dim rangenum As Double

Dim strMaxTicker, strMinTicker, strHighestTicker As String

'create variables for stock prices, cols C-F

Dim dblOpen, dblHigh, dblLow, dblClose As Double

'create variables to hold start and end values for looping through rows

Dim dblStart\_row, dblEnd\_row, dblStart\_rowSummary, dblEnd\_rowSummary As Double

'variable for worksheet

Dim ws As Worksheet

'wrap ENTIRE code in worksheet function to make work on all sheets

For Each ws In ThisWorkbook.Worksheets

'Set up summary area according to assignment parameters

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

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

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

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

'get total number of records (rows) in worksheet

dblStart\_row = 2

dblEnd\_row = ws.Cells(Rows.Count, 1).End(xlUp).Row

'assign starting values to variables

i = 2 'start on row 2 for cols A-G

j = 2 'start on row 2 for cols I-L

dblHigh = ws.Cells(i, 4).Value

dblLow = ws.Cells(i, 5).Value

dblClose = ws.Cells(i, 6).Value

'prepare to iterate through ticker col A, grab ticker and open values

'NOTE: Instructions do not call for using date column to determine

'the first and last dated entry for each ticker. Rather, the data is sorted

'by ticker, then date, so can capture first "open" and last "close" values per ticker

While i < dblEnd\_row

'this is starting point for sub and whenever the ticker changes, resets variables

strTicker = ws.Cells(i, 1).Value 'grabs value from first new ticker row

dblOpen = ws.Cells(i, 3).Value 'grabs value from first new ticker row

dblTotalVolume = 0 'resets volume to zero to start summing for new ticker

'get volume, add row values for each ticker, exit while loop and start over with next ticker

'because volume is totaled up for each ticker, it is done within a loop, comparing ticker,

'and when ticker changes, exit loop, reset volume variable to zero for next ticker.

While strTicker = ws.Cells(i, 1).Value

dblTotalVolume = dblTotalVolume + ws.Cells(i, 7).Value

'continue moving down rows until ticker changes

i = i + 1

Wend

'when ticker changes, the above While Loop will end.

'Need to capture the values, do calculations to populate summary table

dblHigh = ws.Cells(i - 1, 4).Value

dblLow = ws.Cells(i - 1, 5).Value

dblClose = ws.Cells(i - 1, 6).Value

dblYearlyChange = dblClose - dblOpen

dblPercentChange = dblYearlyChange / dblOpen

'Now that ticker changed, assign summary values to cols 9-12. Use variable j for this

ws.Cells(j, 9).Value = strTicker

ws.Cells(i, 9).HorizontalAlignment = xlCenter

ws.Cells(j, 10).Value = dblYearlyChange

ws.Cells(j, 11).Value = dblPercentChange

ws.Cells(j, 12).Value = dblTotalVolume

ws.Cells(i, 9).HorizontalAlignment = xlCenter

'conditional formatting on Yearly Change column, according to pos or neg change

If ws.Cells(j, 10) >= 0 Then

ws.Cells(j, 10).Interior.ColorIndex = "4"

Else

ws.Cells(j, 10).Interior.ColorIndex = "3"

End If

'NOTE conditional formatting, the instructions only say to apply to Yearly Change column,

'but the grading criteria says 10 points for each Yearly Change and Percent Change columns.

'I changed the color scheme for Percent Changed to make it easier to see

'conditional formatting on Percent Change column, according to pos or neg change

If ws.Cells(j, 11) >= 0 Then

ws.Cells(j, 11).Interior.ColorIndex = "50"

Else

ws.Cells(j, 11).Interior.ColorIndex = "38"

End If

'Percentage column format to 2 decimals and % sign

ws.Cells(j, 11).NumberFormat = "0.00%"

'add 1 to j to move to next row of summary table

j = j + 1

Wend

'##################################################################

'add functionality to pull out most pos, neg, and highest total volume into second summary table

'set up second summary table

ws.Cells(1, 16) = "Ticker"

ws.Cells(1, 17) = "Value"

ws.Columns("O:Q").ColumnWidth = 22

ws.Cells(2, 15) = "Greatest % Increase"

ws.Cells(3, 15) = "Greatest % Decrease"

ws.Cells(4, 15) = "Greatest Total Volume"

'determine total number of rows in orig summary table

'will use cols 9-12 to get these values

dblEnd\_rowSummary = ws.Cells(Rows.Count, 9).End(xlUp).Row

dblStart\_rowSummary = 2

'Range("M2").Value = dblEnd\_rowSummary

i = 2

dblMax = 0

dblMin = 0

strMaxTicker = ""

strMinTicker = ""

strHighestTicker = ""

'loop thru percent changed for most pos change

While IsEmpty(ws.Cells(i, 11)) = False

If ws.Cells(i, 11).Value > dblMax Then

dblMax = ws.Cells(i, 11).Value

strMaxTicker = ws.Cells(i, 9).Value

End If

i = i + 1

Wend

'repeat for most neg using MinTicker,Min

i = 2

While IsEmpty(ws.Cells(i, 11)) = False

If ws.Cells(i, 11).Value < dblMin Then

dblMin = ws.Cells(i, 11).Value

strMinTicker = ws.Cells(i, 9).Value

End If

i = i + 1

Wend

'repeat for most Highest Vol using HighestTicker,Highest

i = 2

While IsEmpty(ws.Cells(i, 11)) = False

If ws.Cells(i, 12).Value > dblHighest Then

dblHighest = ws.Cells(i, 12).Value

strHighestTicker = ws.Cells(i, 9).Value

End If

i = i + 1

Wend

'populate table

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

ws.Cells(2, 17).Value = dblMax

ws.Cells(2, 17).NumberFormat = "0.00%"

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

ws.Cells(3, 17).Value = dblMin

ws.Cells(3, 17).NumberFormat = "0.00%"

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

ws.Cells(4, 17).Value = dblHighest

ws.Cells(4, 17).NumberFormat = "##0.00E+0"

'MsgBox ws.Name

'move to next worksheet

Next

End Sub