Option Explicit

' Top‐level: loops every worksheet in the workbook

Sub StockAnalysis()

Dim ws As Worksheet

Application.ScreenUpdating = False

For Each ws In ThisWorkbook.Worksheets

Call AnalyzeQuarter(ws)

Next ws

Application.ScreenUpdating = True

MsgBox "All sheets processed!", vbInformation

End Sub

' Analyze one sheet, using header‐name lookup for flexibility

Sub AnalyzeQuarter(ws As Worksheet)

Dim lastRow As Long, summaryRow As Long, i As Long

Dim curTicker As String, totalVolume As Double

Dim openPrice As Double, closePrice As Double

Dim changeValue As Double, pctChange As Double

' These will hold the column numbers for each field

Dim colTicker As Long, colOpen As Long, colClose As Long, colVol As Long

Dim colSummaryStart As Long

' 1) Find the last row of data

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

' 2) Find your columns by header text (row 1)

With ws.Rows(1)

colTicker = .Find(What:="ticker", LookAt:=xlPart, MatchCase:=False).Column

colOpen = .Find(What:="open", LookAt:=xlPart, MatchCase:=False).Column

colClose = .Find(What:="close", LookAt:=xlPart, MatchCase:=False).Column

colVol = .Find(What:="vol", LookAt:=xlPart, MatchCase:=False).Column

End With

' 3) Set up headers for your summary table, 8 cols to the right of ticker

colSummaryStart = colTicker + 8

With ws

.Cells(1, colSummaryStart + 0).Value = "Ticker"

.Cells(1, colSummaryStart + 1).Value = "Total Volume"

.Cells(1, colSummaryStart + 2).Value = "Quarterly Change"

.Cells(1, colSummaryStart + 3).Value = "Percent Change"

End With

' 4) Initialize running vars for the first ticker

summaryRow = 2

curTicker = ws.Cells(2, colTicker).Value

openPrice = ws.Cells(2, colOpen).Value

totalVolume = ws.Cells(2, colVol).Value

' 5) Loop rows 3→lastRow, break when ticker changes

For i = 3 To lastRow

If ws.Cells(i, colTicker).Value = curTicker Then

totalVolume = totalVolume + ws.Cells(i, colVol).Value

Else

' output the previous ticker

closePrice = ws.Cells(i - 1, colClose).Value

changeValue = closePrice - openPrice

If openPrice <> 0 Then pctChange = changeValue / openPrice Else pctChange = 0

With ws

.Cells(summaryRow, colSummaryStart + 0).Value = curTicker

.Cells(summaryRow, colSummaryStart + 1).Value = totalVolume

.Cells(summaryRow, colSummaryStart + 2).Value = changeValue

.Cells(summaryRow, colSummaryStart + 3).Value = pctChange

.Cells(summaryRow, colSummaryStart + 3).NumberFormat = "0.00%"

End With

' reset for next ticker

summaryRow = summaryRow + 1

curTicker = ws.Cells(i, colTicker).Value

openPrice = ws.Cells(i, colOpen).Value

totalVolume = ws.Cells(i, colVol).Value

End If

Next i

' 6) Output the very last ticker

closePrice = ws.Cells(lastRow, colClose).Value

changeValue = closePrice - openPrice

If openPrice <> 0 Then pctChange = changeValue / openPrice Else pctChange = 0

With ws

.Cells(summaryRow, colSummaryStart + 0).Value = curTicker

.Cells(summaryRow, colSummaryStart + 1).Value = totalVolume

.Cells(summaryRow, colSummaryStart + 2).Value = changeValue

.Cells(summaryRow, colSummaryStart + 3).Value = pctChange

.Cells(summaryRow, colSummaryStart + 3).NumberFormat = "0.00%"

End With

' 7) Apply conditional formatting (green for >0, red for <0)

Dim rngChange As Range, rngPct As Range

Set rngChange = ws.Range(ws.Cells(2, colSummaryStart + 2), ws.Cells(summaryRow, colSummaryStart + 2))

Set rngPct = ws.Range(ws.Cells(2, colSummaryStart + 3), ws.Cells(summaryRow, colSummaryStart + 3))

With rngChange

.FormatConditions.Delete

.FormatConditions.Add Type:=xlCellValue, Operator:=xlGreater, Formula1:="=0"

.FormatConditions(1).Interior.Color = vbGreen

.FormatConditions.Add Type:=xlCellValue, Operator:=xlLess, Formula1:="=0"

.FormatConditions(2).Interior.Color = vbRed

End With

With rngPct

.FormatConditions.Delete

.FormatConditions.Add Type:=xlCellValue, Operator:=xlGreater, Formula1:="=0"

.FormatConditions(1).Interior.Color = vbGreen

.FormatConditions.Add Type:=xlCellValue, Operator:=xlLess, Formula1:="=0"

.FormatConditions(2).Interior.Color = vbRed

End With

' 8) Find Greatest % Inc/Dec and Greatest Volume

Dim maxInc As Double, maxDec As Double, maxVol As Double

Dim tkrInc As String, tkrDec As String, tkrVol As String

Dim r As Long

' seed with row 2

maxInc = ws.Cells(2, colSummaryStart + 3).Value

maxDec = maxInc

maxVol = ws.Cells(2, colSummaryStart + 1).Value

tkrInc = ws.Cells(2, colSummaryStart + 0).Value

tkrDec = tkrInc

tkrVol = tkrInc

For r = 2 To summaryRow

If ws.Cells(r, colSummaryStart + 3).Value > maxInc Then

maxInc = ws.Cells(r, colSummaryStart + 3).Value

tkrInc = ws.Cells(r, colSummaryStart + 0).Value

End If

If ws.Cells(r, colSummaryStart + 3).Value < maxDec Then

maxDec = ws.Cells(r, colSummaryStart + 3).Value

tkrDec = ws.Cells(r, colSummaryStart + 0).Value

End If

If ws.Cells(r, colSummaryStart + 1).Value > maxVol Then

maxVol = ws.Cells(r, colSummaryStart + 1).Value

tkrVol = ws.Cells(r, colSummaryStart + 0).Value

End If

Next r

' 9) Output those three metrics over by 12 columns

With ws

.Cells(1, colSummaryStart + 12).Value = "Metric"

.Cells(1, colSummaryStart + 13).Value = "Ticker"

.Cells(1, colSummaryStart + 14).Value = "Value"

.Cells(2, colSummaryStart + 12).Value = "Greatest % Increase"

.Cells(2, colSummaryStart + 13).Value = tkrInc

.Cells(2, colSummaryStart + 14).Value = maxInc

.Cells(3, colSummaryStart + 12).Value = "Greatest % Decrease"

.Cells(3, colSummaryStart + 13).Value = tkrDec

.Cells(3, colSummaryStart + 14).Value = maxDec

.Cells(4, colSummaryStart + 12).Value = "Greatest Total Volume"

.Cells(4, colSummaryStart + 13).Value = tkrVol

.Cells(4, colSummaryStart + 14).Value = maxVol

.Range(.Cells(2, colSummaryStart + 14), .Cells(3, colSummaryStart + 14)).NumberFormat = "0.00%"

End With

End Sub