**Module 2 Challenge – Notes/Brain Dump**

**To add VBA files to GitHub, you can follow these general steps:**

1. Make sure you have a GitHub account and have set up a repository for your project.
2. Open your command line interface (e.g., Git Bash).
3. Navigate to the directory where your VBA files are located using the cd command.
4. Initialize a new Git repository in the directory by running git init.
5. Add the VBA files to the staging area using git add <file> or git add . to add all files.
6. Commit the changes with a message using git commit -m "Your commit message here".
7. Link your local repository to your GitHub repository using git remote add origin <repository URL>.
8. Push your changes to GitHub using git push -u origin master.

Remember to replace <file> and <repository URL> with the actual file names and repository URL. If you encounter any issues during this process, please let me know so I can assist you further.

**To save changes to a GitHub repository without changing the main branch (formerly known as the master branch), you can follow these steps:**

1. **Create a New Branch:**
   * Start by creating a new branch in your local repository. You can do this by running the command git checkout -b new-branch-name.
2. **Make Changes:**
   * Make the necessary changes to your files in the new branch.
3. **Add and Commit Changes:**
   * Add the changes to the staging area using git add . or git add <file> for specific files.
   * Commit the changes using git commit -m "Your commit message".
4. **Push Changes to GitHub:**
   * Push the changes to the new branch on GitHub using git push origin new-branch-name.

By following these steps, you can save your changes to a separate branch in your GitHub repository without affecting the main branch.

**To change the branch name in GitHub, you can follow these steps:**

1. **Rename the Branch Locally:**
   * Start by renaming the branch locally on your machine. You can do this using the following command:
   * git branch -m old-branch-name new-branch-name
   * This command renames the branch from old-branch-name to new-branch-name.
2. **Push the Renamed Branch to GitHub:**
   * Push the renamed branch to GitHub using the following command:
   * git push origin new-branch-name
   * This command will push the renamed branch to GitHub with the new name.
3. **Delete the Old Branch on GitHub (Optional):**
   * If you want to delete the old branch on GitHub, you can do so using the following command:
   * git push origin --delete old-branch-name
   * This command will delete the old branch from the remote repository on GitHub.

By following these steps, you can rename a branch locally and then push the renamed branch to GitHub with the new name.

**In VBA, there are several data types that you can use to define the kind of data that a variable can hold. Here are some common data types in VBA and when you might use them:**

1. **String**: Used to store text or a combination of text and numbers. For example, storing names, addresses, or any textual information.
2. **Integer**: Used to store whole numbers without decimal points. For example, counting items or representing quantities that can only be whole numbers.
3. **Long**: Similar to Integer but can hold larger numbers. Use Long when you need to store very large whole numbers.
4. **Double**: Used to store numbers with decimal points. For example, storing measurements, percentages, or any numeric values that require decimal precision.
5. **Boolean**: Used to store logical values - True or False. For example, when you need to represent conditions or states that are either true or false.
6. **Date**: Used to store date and time values. For example, recording timestamps, scheduling events, or any date-related information.
7. **Variant**: A versatile data type that can store any type of data. It's flexible but can lead to potential issues if not used carefully. It's better to use specific data types when possible to avoid unexpected behavior.

When choosing a data type, consider the range of values you need to store, the precision required, and the type of operations you will perform on the data. Using the appropriate data type can help optimize memory usage and ensure data integrity in your VBA code.

Sub stock\_data():

Dim Ticker As Variable

Dim Qtly\_Change As Double

Dim Opening\_Price As Double

Dim Closing\_Price As Double

Dim Percent\_Change As Double

Dim Total\_Stock\_Vol As Long

Dim LastRow As Long

Dim ws As Worksheet

For Each ws In Worksheets

' overall goal for this part: Must identify each ticker for each worksheet; add column with each ticker for each worksheet and give new column name of Ticker.

Dim i As Long

'find last row for each ws

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

' Loop through each row in the column and copy the value to another column

For i = 2 To LastRow

ws.Cells(i, 1).Value = Ticker

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

Next i

'add Ticker as column title

ws.Cells(1, 9).Value = "Ticker"

Next ws

End Sub

**Had a lot of issues**

**debug Sub stock\_data() Dim Ticker As Variant Dim Qtly\_Change As Variant Dim Opening\_Price As Variant Dim Closing\_Price As Variant Dim Percent\_Change As Variant Dim Total\_Stock\_Vol As Long Dim LastRow As Long Dim ws As Worksheet For Each ws In Worksheets ' overall goal for this part: Must identify each ticker for each worksheet; add column with each ticker for each worksheet and give new column name of Ticker. Dim i As Long 'find last row for each ws LastRow = ws.Cells(Rows.Count, 1).End(xlUp).Row ' Loop through each row in the column and copy the value to another column For i = 2 To LastRow ws.Cells(i, 1).Value = Ticker ws.Cells(i, 9).Value = Ticker Next i 'add Ticker as column title ws.Cells(1, 9).Value = "Ticker" Next ws End Sub**

It seems like there are a few issues in your VBA script. Let's address them:

1. You are not assigning any values to the Ticker variable before using it to assign values to cells. You need to assign the ticker value from the worksheet to the Ticker variable inside the loop.
2. You are assigning the same Ticker value to all cells in column 1 and column 9. You should assign the correct ticker value to the respective rows in column 9.

Here is an updated version of your script:

Sub stock\_data()

Dim Ticker As Variant

Dim Qtly\_Change As Variant

Dim Opening\_Price As Variant

Dim Closing\_Price As Variant

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Long

Dim LastRow As Long

Dim ws As Worksheet

For Each ws In Worksheets

' overall goal for this part: Must identify each ticker for each worksheet; add column with each ticker for each worksheet and give new column name of Ticker.

Dim i As Long

'find last row for each ws

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

' Loop through each row in the column and copy the value to another column

For i = 2 To LastRow

Ticker = ws.Cells(i, 1).Value ' Assign the ticker value from the worksheet to the Ticker variable

ws.Cells(i, 9).Value = Ticker ' Assign the Ticker value to column 9 for each row

Next i

'add Ticker as column title

ws.Cells(1, 9).Value = "Ticker"

Next ws

End Sub

Please try this updated script and let me know if you encounter any further issues.

***Code so far:***

***Sub stock\_data()***

***Dim Ticker As Variant***

***Dim Qtly\_Change As Variant***

***Dim Opening\_Price As Variant***

***Dim Closing\_Price As Variant***

***Dim Percent\_Change As Variant***

***Dim Total\_Stock\_Vol As Long***

***Dim LastRow As Long***

***Dim ws As Worksheet***

***For Each ws In Worksheets***

***' overall goal for this part: Must identify each ticker for each worksheet; add column with each ticker for each worksheet and give new column name of Ticker.***

***Dim i As Long***

***'find last row for each ws***

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

***' Copy the first value to the new column***

***newRow = 2***

***ws.Cells(newRow, "I").Value = ws.Cells(2, "A").Value***

***currentValue = ws.Cells(2, "A").Value***

***' Loop through the rows to consolidate values***

***For i = 2 To LastRow***

***If ws.Cells(i, "A").Value <> currentValue Then***

***newRow = newRow + 1***

***ws.Cells(newRow, "I").Value = ws.Cells(i, "A").Value***

***currentValue = ws.Cells(i, "A").Value***

***End If***

***Next i***

***'add Ticker as column title***

***ws.Cells(1, 9).Value = "Ticker"***

***'goal for this part: calculate qtly change for each ticker) and populate it into column J. Column J with own title***

***'For i = 2 To LastRow***

***'Qtly\_Change = []***

***Next ws***

***End Sub***

Had issues so tried the below

Sub stock\_data()

Dim Ticker As Variant

Dim Qtly\_Change As Variant

Dim Opening\_Price As Variant

Dim Closing\_Price As Variant

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Long

Dim LastRow As Long

Dim ws As Worksheet

For Each ws In Worksheets

' overall goal for this part: Must identify each ticker for each worksheet; add column with each ticker for each worksheet and give new column name of Ticker.

Dim i As Long

'find last row for each ws

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

' Copy the first ticker value to the new column

tickernewrow = 2

ws.Cells(tickernewrow, "I").Value = ws.Cells(2, "A").Value

currenttickerValue = ws.Cells(2, "A").Value

' Loop through the rows to consolidate values

For i = 2 To LastRow

If ws.Cells(i, "A").Value <> currenttickerValue Then

tickernewrow = tickernewrow + 1

ws.Cells(tickernewrow, "I").Value = ws.Cells(i, "A").Value

currenttickerValue = ws.Cells(i, "A").Value

'goal for this part: calculate qtly change for each ticker) and populate it into column J. Column J with own title

Opening\_Price = ws.Cells(tickernewrow, "C").Value

Closing\_Price = ws.Cells(LastRow, "C").Value

Qtly\_Change = Closing\_Price - Opening\_Price

ws.Cells(tickernewrow, "J").Value = Qtly\_Change

End If

Next i

'add Ticker as column title

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quaterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

Next ws

End Sub

This populates the incorrect Qtly\_Change amounts. I put the last row of the whole ws as Closing which is wrong. I need the amount that is the last instance of that ticker value.

**code for price in a column associated with the last instance of a ticker id in a different column**

To retrieve the price in a column associated with the last instance of a ticker ID in a different column, you can iterate through the rows in reverse order and find the last occurrence of the ticker ID. Once you find the last occurrence, you can retrieve the price associated with that row.

Here is a sample VBA code snippet that demonstrates this concept:

Sub GetLastPriceForTicker()

Dim ws As Worksheet

Dim LastRow As Long

Dim TickerID As String

Dim LastPrice As Double

' Specify the worksheet where you want to search

Set ws = ThisWorkbook.Worksheets("Sheet1")

' Find the last row in the worksheet

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

' Specify the ticker ID you are interested in

TickerID = "ABC" ' Replace "ABC" with the actual ticker ID

' Loop through the rows in reverse order to find the last occurrence of the ticker ID

For i = LastRow To 2 Step -1

If ws.Cells(i, "A").Value = TickerID Then

' Retrieve the price associated with the last occurrence of the ticker ID

LastPrice = ws.Cells(i, "B").Value ' Assuming price is in column B

Exit For ' Exit the loop once the last occurrence is found

End If

Next i

' Display the last price associated with the ticker ID

MsgBox "The last price for ticker " & TickerID & " is: " & LastPrice

End Sub

In this code snippet:

* Replace "Sheet1" with the actual name of the worksheet you are working with.
* Replace "ABC" with the specific ticker ID you are interested in.
* Adjust the column letter in ws.Cells(i, "B").Value to match the column where the price is located.

This code will search for the last occurrence of a specified ticker ID in column A and retrieve the price associated with that last occurrence from column B.

***I turned that into the below***

Sub stock\_data()

Dim Ticker As Variant

Dim Qtly\_Change As Variant

Dim Opening\_Price As Variant

Dim Closing\_Price As Variant

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Long

Dim LastRow As Long

Dim ws As Worksheet

For Each ws In Worksheets

' overall goal for this part: Must identify each ticker for each worksheet; add column with each ticker for each worksheet and give new column name of Ticker.

Dim i As Long

'find last row for each ws

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

' Copy the first ticker value to the new column   
tickernewrow = 2

ws.Cells(tickernewrow, "I").Value = ws.Cells(2, "A").Value

currenttickerValue = ws.Cells(2, "A").Value

' Loop through the rows to consolidate values

For i = 2 To LastRow

If ws.Cells(i, "A").Value <> currenttickerValue Then

tickernewrow = tickernewrow + 1

ws.Cells(tickernewrow, "I").Value = ws.Cells(i, "A").Value

currenttickerValue = ws.Cells(i, "A").Value

'goal for this part: calculate qtly change for each ticker) and populate it into column J. Column J with own title

For j = LastRow To 2 Step -1

If ws.Cells(j, "A").Value = currenttickerValue Then

' Retrieve the price associated with the last occurrence of the ticker ID

Closing\_Price = ws.Cells(j, "B").Value ' Assuming price is in column B

Opening\_Price = ws.Cells(tickernewrow, "C").Value

Qtly\_Change = Closing\_Price - Opening\_Price

ws.Cells(tickernewrow, "J").Value = Qtly\_Change

End If

Next j

Exit For ' Exit the loop once the last occurrence is found

End If

Next i

'add Ticker as column title

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quaterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

Next ws

End Sub

I tried to fix it using the above as a template. This only populates 2 cells in Qtly Change column and they are wrong (has date and time). Exiting the loop too early and calc’ing wrong thing?

**Asked for a debug**

I see a couple of issues in your VBA script. Let's address them:

1. The variable j used in the inner loop to find the last occurrence of the ticker ID is not declared. You need to declare it before using it in the loop.
2. You are exiting the outer loop prematurely when finding the first occurrence of a different ticker ID. This might not be the desired behavior as you may want to calculate quarterly changes for all occurrences of each ticker.

Here is the corrected code snippet with the necessary adjustments:

Sub stock\_data()

Dim Ticker As Variant

Dim Qtly\_Change As Variant

Dim Opening\_Price As Variant

Dim Closing\_Price As Variant

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Long

Dim LastRow As Long

Dim ws As Worksheet

For Each ws In Worksheets

' Find the last row for each worksheet

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

' Copy the first ticker value to the new column

Dim tickernewrow As Long

tickernewrow = 2

ws.Cells(tickernewrow, "I").Value = ws.Cells(2, "A").Value

Dim currenttickerValue As Variant

currenttickerValue = ws.Cells(2, "A").Value

' Loop through the rows to consolidate values

Dim i As Long

For i = 2 To LastRow

If ws.Cells(i, "A").Value <> currenttickerValue Then

tickernewrow = tickernewrow + 1

ws.Cells(tickernewrow, "I").Value = ws.Cells(i, "A").Value

currenttickerValue = ws.Cells(i, "A").Value

' Find the last occurrence of the current ticker ID

Dim j As Long

For j = LastRow To 2 Step -1

If ws.Cells(j, "A").Value = currenttickerValue Then

Closing\_Price = ws.Cells(j, "B").Value ' Assuming price is in column B

Opening\_Price = ws.Cells(tickernewrow, "C").Value

Qtly\_Change = Closing\_Price - Opening\_Price

ws.Cells(tickernewrow, "J").Value = Qtly\_Change

Exit For ' Exit the loop once the last occurrence is found

End If

Next j

End If

Next i

' Add column titles

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quarterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

Next ws

End Sub

These corrections should help you calculate the quarterly changes for each ticker ID and populate them into the respective columns in your VBA script.

***Tried the below as adjustment***

Sub stock\_data()

Dim Ticker As Variant

Dim Qtly\_Change As Variant

Dim Opening\_Price As Variant

Dim Closing\_Price As Variant

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Long

Dim LastRow As Long

Dim ws As Worksheet

For Each ws In Worksheets

' overall goal for this part: Must identify each ticker for each worksheet; add column with each ticker for each worksheet and give new column name of Ticker.

Dim i As Long

'find last row for each ws

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

' Copy the first ticker value to the new column

tickernewrow = 2

ws.Cells(tickernewrow, "I").Value = ws.Cells(2, "A").Value

currenttickerValue = ws.Cells(2, "A").Value

' Loop through the rows to consolidate values

For i = 2 To LastRow

If ws.Cells(i, "A").Value <> currenttickerValue Then

tickernewrow = tickernewrow + 1

ws.Cells(tickernewrow, "I").Value = ws.Cells(i, "A").Value

currenttickerValue = ws.Cells(i, "A").Value

'goal for this part: calculate qtly change for each ticker) and populate it into column J. Column J with own title

' Find the last occurrence of the current ticker ID

Dim j As Long

For j = LastRow To 2 Step -1

If ws.Cells(j, "A").Value = currenttickerValue Then

Closing\_Price = ws.Cells(j, "F").Value

Opening\_Price = ws.Cells(tickernewrow, "C").Value

Qtly\_Change = Closing\_Price - Opening\_Price

ws.Cells(tickernewrow, "J").Value = Qtly\_Change

Exit For ' Exit the loop once the last occurrence is found

End If

Next j

End If

Next i

' Add column titles

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quarterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

Next ws

End Sub

Populates in Qtrly Change but missing first non-column cell and results are way too high (ex. 41, 107, etc). Should be 5, 0.05 etc for beginning values.

**Asked Ai for debug help**

t seems like there are a couple of issues in your VBA script. Let's address them:

1. **Populating Column J (Quarterly Change):**
   * The code to calculate the quarterly change and populate it into column J seems to be inside the loop where you are identifying and populating the ticker symbols. This part should be outside that loop to ensure that the quarterly change is calculated for each row after all ticker symbols have been identified.
   * Move the section that calculates the quarterly change and populates it into column J outside the ticker identification loop.
2. **Column Titles:**
   * You are setting the column titles after the loop that processes the data. It's better to set the column titles before processing the data to ensure they are present when the data is populated.
   * Move the code that sets the column titles to the beginning of the loop, before processing the data.

Here is the updated version of your script with the changes applied:

Sub stock\_data()

Dim Ticker As Variant

Dim Qtly\_Change As Variant

Dim Opening\_Price As Variant

Dim Closing\_Price As Variant

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Long

Dim LastRow As Long

Dim ws As Worksheet

For Each ws In Worksheets

' Add column titles

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quarterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

' overall goal for this part: Must identify each ticker for each worksheet; add column with each ticker for each worksheet and give new column name of Ticker.

Dim i As Long

'find last row for each ws

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

' Copy the first ticker value to the new column

tickernewrow = 2

ws.Cells(tickernewrow, "I").Value = ws.Cells(2, "A").Value

currenttickerValue = ws.Cells(2, "A").Value

' Loop through the rows to consolidate values

For i = 2 To LastRow

If ws.Cells(i, "A").Value <> currenttickerValue Then

tickernewrow = tickernewrow + 1

ws.Cells(tickernewrow, "I").Value = ws.Cells(i, "A").Value

currenttickerValue = ws.Cells(i, "A").Value

End If

Next i

' Calculate Quarterly Change and Populate Column J

For i = 2 To LastRow

currenttickerValue = ws.Cells(i, "A").Value

Closing\_Price = ws.Cells(i, "F").Value

Opening\_Price = ws.Cells(i, "C").Value

Qtly\_Change = Closing\_Price - Opening\_Price

ws.Cells(i, "J").Value = Qtly\_Change

Next i

Next ws

End Sub

Please try running this updated script and let me know if you encounter any more issues.

Sub stock\_data()

Dim Ticker As Variant

Dim Qtly\_Change As Variant

Dim Opening\_Price As Variant

Dim Closing\_Price As Variant

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Long

Dim LastRow As Long

Dim ws As Worksheet

For Each ws In Worksheets

' Add column titles

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quarterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

' overall goal for this part: Must identify each ticker for each worksheet; add column with each ticker for each worksheet and give new column name of Ticker.

Dim i As Long

'find last row for each ws

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

' Copy the first ticker value to the new column

tickernewrow = 2

ws.Cells(tickernewrow, "I").Value = ws.Cells(2, "A").Value

currenttickerValue = ws.Cells(2, "A").Value

' Loop through the rows to consolidate values

For i = 2 To LastRow

If ws.Cells(i, "A").Value <> currenttickerValue Then

tickernewrow = tickernewrow + 1

ws.Cells(tickernewrow, "I").Value = ws.Cells(i, "A").Value

currenttickerValue = ws.Cells(i, "A").Value

End If

Next i

' Calculate Quarterly Change and Populate Column J

For i = 2 To LastRow

currenttickerValue = ws.Cells(i, "A").Value

Closing\_Price = ws.Cells(i, "F").Value

Opening\_Price = ws.Cells(i, "C").Value

Qtly\_Change = Closing\_Price - Opening\_Price

ws.Cells(i, "J").Value = Qtly\_Change

Next i

Next ws

End Sub

Better but still incorrect values.

***Tried to adjust some stuff***

Sub stock\_data()

Dim Ticker As Variant

Dim Qtly\_Change As Variant

Dim Opening\_Price As Variant

Dim Closing\_Price As Variant

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Long

Dim LastRow As Long

Dim ws As Worksheet

For Each ws In Worksheets

' Add column titles

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quarterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

' overall goal for this part: Must identify each ticker for each worksheet; add column with each ticker for each worksheet and give new column name of Ticker.

Dim i As Long

'find last row for each ws

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

' Copy the first ticker value to the new column

Dim tickernewrow As Variant

tickernewrow = 2

ws.Cells(tickernewrow, "I").Value = ws.Cells(2, "A").Value

Dim currenttickervalue As Variant

Dim currenttickerrow As Variant

currenttickervalue = ws.Cells(2, "A").Value

' Loop through the rows to consolidate values

For i = 2 To LastRow

If ws.Cells(i, "A").Value <> currenttickervalue Then

tickernewrow = tickernewrow + 1

ws.Cells(tickernewrow, "I").Value = ws.Cells(i, "A").Value

currenttickervalue = ws.Cells(i, "A").Value

currenttickerrow = i 'making i the row where the tickervalue changes

' Calculate Quarterly Change and Populate Column J

For j = currenttickerrow To LastRow

Closing\_Price = ws.Cells(j, "F").Value

Opening\_Price = ws.Cells(i, "C").Value

Qtly\_Change = Closing\_Price - Opening\_Price

ws.Cells(j, "J").Value = Qtly\_Change

End If

Next i

Next j

Next ws

End Sub

***De-bugged before running bc I was struggling with order of I and j loops and when to close each.***

**I see a couple of issues in your VBA script. Let's address them:**

1. The For j loop is not properly closed. You need to add an End If statement to close the If condition before the Next i statement.
2. The Next j statement is not correctly placed. It should be after the End If statement to close the For j loop.

Here is the corrected version of your VBA script:

Sub stock\_data()

Dim Ticker As Variant

Dim Qtly\_Change As Variant

Dim Opening\_Price As Variant

Dim Closing\_Price As Variant

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Long

Dim LastRow As Long

Dim ws As Worksheet

For Each ws In Worksheets

' Add column titles

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quarterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

' overall goal for this part: Must identify each ticker for each worksheet; add column with each ticker for each worksheet and give new column name of Ticker.

Dim i As Long

'find last row for each ws

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

' Copy the first ticker value to the new column

Dim tickernewrow As Variant

tickernewrow = 2

ws.Cells(tickernewrow, "I").Value = ws.Cells(2, "A").Value

Dim currenttickervalue As Variant

Dim currenttickerrow As Variant

currenttickervalue = ws.Cells(2, "A").Value

' Loop through the rows to consolidate values

For i = 2 To LastRow

If ws.Cells(i, "A").Value <> currenttickervalue Then

tickernewrow = tickernewrow + 1

ws.Cells(tickernewrow, "I").Value = ws.Cells(i, "A").Value

currenttickervalue = ws.Cells(i, "A").Value

currenttickerrow = i 'making i the row where the tickervalue changes

' Calculate Quarterly Change and Populate Column J

For j = currenttickerrow To LastRow

Closing\_Price = ws.Cells(j, "F").Value

Opening\_Price = ws.Cells(i, "C").Value

Qtly\_Change = Closing\_Price - Opening\_Price

ws.Cells(j, "J").Value = Qtly\_Change

Next j

End If

Next i

Next ws

End Sub

This is wrong bc need in the j loop a way to find the open price at the first instance of the currenttickervalue

***Difficulty calculating currenttickerstart in below***

Sub stock\_data()

Dim Ticker As Variant

Dim Qtly\_Change As Variant

Dim Opening\_Price As Variant

Dim Closing\_Price As Variant

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Long

Dim LastRow As Long

Dim ws As Worksheet

For Each ws In Worksheets

' Add column titles

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quarterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

' overall goal for this part: Must identify each ticker for each worksheet; add column with each ticker for each worksheet and give new column name of Ticker.

Dim i As Long

'find last row for each ws

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

' Copy the first ticker value to the new column

Dim tickernewrow As Variant

tickernewrow = 2

ws.Cells(tickernewrow, "I").Value = ws.Cells(2, "A").Value

Dim currenttickervalue As Variant

Dim currenttickerlast As Variant

'Dim newtickerstart As Variant

Dim currenttickerstart As Variant

'loop to check if current ticker value is the same as the one before it

' previous simple ver currenttickervalue = ws.Cells(2, "A").Value

' Loop through rows starting from the second row

For i = 2 To LastRow

' Compare values in current row with values in the row before it

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

' Loop through the rows to consolidate values

tickernewrow = tickernewrow + 1

ws.Cells(tickernewrow, "I").Value = ws.Cells(i, "A").Value

currenttickervalue = ws.Cells(i, "A").Value

'currenttickerstart = [] 'need first instance of current ticker all being compared to

currenttickerlast = i - 1 'to use for last instance of the ticker I was comparing everything to

'newtickerstart = i ' to use as 1st instance of the new ticker

' Calculate Quarterly Change and Populate Column J

Closing\_Price = ws.Cells(currenttickerlast, "F").Value

Opening\_Price = ws.Cells(currenttickerstart, "C").Value

Qtly\_Change = Closing\_Price - Opening\_Price

ws.Cells(tickernewrow, "J").Value = Qtly\_Change

End If

Next i

Next ws

End Sub

**Help with calc currenttickerstart 🡪 not very helpful. Will have to calc for 1st unqique ticker ID’s qty change and the do for loop for rest of unique ticker IDs.**

**what can you do with vba for loops aside from conditionals**=

In addition to using VBA for loops for conditionals, you can leverage them for various other tasks, such as:

1. **Iterating Through Data**: You can use for loops to iterate through arrays, ranges, or collections of data to perform calculations, comparisons, or other operations on each element.
2. **Data Manipulation**: for loops are handy for manipulating data within Excel spreadsheets. You can update cell values, format cells, insert or delete rows, and perform other data manipulation tasks.
3. **Automating Repetitive Tasks**: If you have a task that needs to be repeated multiple times with slight variations, for loops can help automate these repetitive actions efficiently.
4. **Generating Reports**: By using for loops, you can extract data from different sources, process it, and generate reports automatically without manual intervention.
5. **Creating Patterns**: You can use for loops to create specific patterns or sequences of data, which can be useful in various applications like generating charts, tables, or visualizations.
6. **Error Checking and Handling**: for loops can be used to check for errors in data or processes and handle them appropriately within your VBA code.

Overall, for loops in VBA are versatile tools that allow you to streamline your coding process, automate tasks, and efficiently work with data in Excel.

**help fix code. trying to set first Opening\_Price output to specific cell value C2 then calculating following Opening\_Price and Closing\_Price based on when i changes (showing new ticker value). Not sure how to create 2nd loop to calculate Qtly\_Change for each time ticker value is different (each time i value meets the conditional). Sub stock\_data() Dim Ticker As Variant Dim Qtly\_Change As Variant Dim Opening\_Price As Variant Dim Closing\_Price As Variant Dim Percent\_Change As Variant Dim Total\_Stock\_Vol As Long Dim LastRow As Long Dim ws As Worksheet For Each ws In Worksheets ' Add column titles ws.Cells(1, 9).Value = "Ticker" ws.Cells(1, 10).Value = "Quarterly Change" ws.Cells(1, 11).Value = "Percent Change" ws.Cells(1, 12).Value = "Total Stock Volume" ' overall goal for this part: Must identify each ticker for each worksheet; add column with each ticker for each worksheet and give new column name of Ticker. Dim i As Long 'find last row for each ws LastRow = ws.Cells(Rows.Count, 1).End(xlUp).Row ' Copy the first ticker value to the new column Dim tickernewrow As Variant tickernewrow = 2 ws.Cells(tickernewrow, "I").Value = ws.Cells(2, "A").Value Dim currenttickervalue As Variant Dim newtickerlastrow As Long Dim newtickerstart As Long 'loop to check if current ticker value is the same as the one before it ' previous simple ver currenttickervalue = ws.Cells(2, "A").Value ' Loop through rows starting from the second row For i = 2 To LastRow ' Compare values in current row with values in the row before it If ws.Cells(i, 1).Value <> ws.Cells(i - 1, 1).Value Then ' Loop through the rows to consolidate values tickernewrow = tickernewrow + 1 ws.Cells(tickernewrow, "I").Value = ws.Cells(i, "A").Value currenttickervalue = ws.Cells(i, "A").Value For j = i To LastRow newtickerlastrow = i - 1 'store last instance of current ticker newtickerstart = i 'store 1st instance of the new ticker 'first qty chng for I2 'continue working from here. need to calc I2's qty change and then formula for next ticker IDs' qty changes (as a loop = cnoditional?) currenttickerstart = 2 Closing\_Price = ws.Cells(currenttickerlast, "F").Value Opening\_Price = ws.Cells(2, "C").Value Qtly\_Change = Closing\_Price - Opening\_Price ws.Cells(tickernewrow, "J").Value = Qtly\_Change currenttickerstart = 'rest of tickers calcs ' Calculate Quarterly Change and Populate Column J Closing\_Price = ws.Cells(currenttickerlast, "F").Value Opening\_Price = ws.Cells(currenttickerstart, "C").Value Qtly\_Change = Closing\_Price - Opening\_Price ws.Cells(tickernewrow, "J").Value = Qtly\_Change End If Next i Next ws**

It looks like you are on the right track with your VBA script for analyzing stock data. I see that you are trying to loop through the rows to calculate the quarterly change for each ticker symbol. Here are a few suggestions to help you fix your code:

1. **Define and Initialize Variables**: Make sure to define and initialize all your variables at the beginning of your script. This helps avoid any potential issues with variable scope and ensures that each variable has a clear purpose.
2. **Use Meaningful Variable Names**: Try to use meaningful variable names to make your code more readable and easier to understand. For example, instead of i, j, ws, consider using names like rowIndex, tickerIndex, worksheet, etc.
3. **Correct Looping Logic**: It seems like you are trying to loop through the rows to calculate the quarterly change for each ticker symbol. You need to ensure that your loop logic is correctly structured to achieve this.
4. **Calculate Quarterly Change**: Calculate the quarterly change for each ticker symbol by subtracting the opening price from the closing price. Make sure to update the opening price whenever a new ticker symbol is encountered.
5. **Update Cell References**: Double-check your cell references to ensure they are pointing to the correct cells for each calculation.

Here is a modified version of your code with some adjustments:

Sub stock\_data()

Dim Ticker As Variant

Dim Qtly\_Change As Variant

Dim Opening\_Price As Variant

Dim Closing\_Price As Variant

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Long

Dim LastRow As Long

Dim ws As Worksheet

For Each ws In Worksheets

' Add column titles

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quarterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

' Find the last row for each worksheet

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

Dim tickerNewRow As Long

tickerNewRow = 2

' Copy the first ticker value to the new column

ws.Cells(tickerNewRow, "I").Value = ws.Cells(2, "A").Value

Dim currentTickerValue As Variant

Dim currentTickerStart As Long

Dim currentTickerLast As Long

' Loop through rows starting from the second row

For i = 2 To LastRow

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

tickerNewRow = tickerNewRow + 1

ws.Cells(tickerNewRow, "I").Value = ws.Cells(i, "A").Value

currentTickerValue = ws.Cells(i, "A").Value

currentTickerStart = i

currentTickerLast = i - 1

' Calculate Quarterly Change and Populate Column J

Closing\_Price = ws.Cells(currentTickerLast, "F").Value

Opening\_Price = ws.Cells(currentTickerStart, "C").Value

Qtly\_Change = Closing\_Price - Opening\_Price

ws.Cells(tickerNewRow, "J").Value = Qtly\_Change

End If

Next i

Next ws

End Sub

***Still issue with properly defining the starting row of current ticker. And the output is limited too.***

**in above code, how to store how many i values looped through to get to current i value that does not meet condition**

To store how many i values have looped through to get to the current i value that does not meet a condition, you can introduce a separate variable to keep track of this count. You can increment this count every time the condition is not met. Here's an example of how you can modify the code to achieve this:

' Create a variable to hold the counter

Dim i As Integer

Dim count As Integer

' Initialize count to 0

count = 0

' Loop through first 10 rows

For i = 1 To 10

' Check if the number is divisible by 2

If i Mod 2 = 0 Then

' Enter "Even Row" in the adjacent cell

Cells(i, 2).Value = "Even Row"

Else

' Enter "Odd Row" in the adjacent cell

Cells(i, 2).Value = "Odd Row"

' Increment the count every time the number is odd

count = count + 1

End If

Next i

' Print the count of odd numbers that were encountered

MsgBox "Count of odd numbers: " & count

In this modified code, the count variable is initialized to 0 before the loop starts. Inside the loop, whenever the condition for odd numbers is met, the count variable is incremented by 1. After the loop completes, you can display the count of odd numbers encountered using a message box or store it for further processing.

***Tried the below. Keeps having errors***

Sub stock\_alphabet()

Dim Ticker As Variant

Dim Qtly\_Change As Variant

Dim Opening\_Price As Variant

Dim Closing\_Price As Variant

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Long

Dim LastRow As Long

Dim ws As Worksheet

Dim currentTickerStart As Long

Dim currentTickerValue As Variant

Dim newtickerstart As Long

Dim currenttickerlast As Long

Dim currenttickerstartrow As Long

Dim tickerNewRow As Long

For Each ws In Worksheets

' Add column titles

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quarterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

' Find the last row for each worksheet

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

tickerNewRow = 2

currentTickerStart = 0

' Copy the first ticker value to the new column

ws.Cells(tickerNewRow, "I").Value = ws.Cells(2, "A").Value

' Loop through rows starting from the second row. define more variables will need

For i = 2 To LastRow

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

tickerNewRow = tickerNewRow + 1

ws.Cells(tickerNewRow, "I").Value = ws.Cells(i, "A").Value

currentTickerValue = ws.Cells(i, "A").Value

currentTickerStart = currentTickerStart + 1 'add 1 for each loop completed of i until not meet condition

currenttickerstartrow = tickerNewRow - currentTickerStart 'use this calc to try and pinpoint where starting row for this ticker is

newtickerstart = i

currenttickerlast = i - 1

' Calculate Quarterly Change and Populate Column J

Closing\_Price = ws.Cells(currenttickerlast, "F").Value

Opening\_Price = ws.Cells(currenttickerstartrow, "C").Value

Qtly\_Change = Closing\_Price - Opening\_Price

ws.Cells(tickerNewRow, "J").Value = Qtly\_Change

End If

Next i

Next ws

End Sub

***Asked for debug help***

Sub stock\_alphabet()

Dim Ticker As Variant

Dim Qtly\_Change As Double

Dim Opening\_Price As Double

Dim Closing\_Price As Double

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Long

Dim LastRow As Long

Dim ws As Worksheet

Dim currentTickerStart As Long

Dim currentTickerValue As Variant

Dim newtickerstart As Long

Dim currenttickerstartrow As Long

For Each ws In Worksheets

' Add column titles

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quarterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

' Find the last row for each worksheet

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

tickerNewRow = 2

currentTickerStart = 0

' Copy the first ticker value to the new column

ws.Cells(tickerNewRow, "I").Value = ws.Cells(2, "A").Value

' Loop through rows starting from the second row. define more variables will need

For i = 2 To LastRow

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

tickerNewRow = tickerNewRow + 1

ws.Cells(tickerNewRow, "I").Value = ws.Cells(i, "A").Value

currentTickerValue = ws.Cells(i, "A").Value

currentTickerStart = currentTickerStart + 1 'add 1 for each loop completed of i until not meet condition

currenttickerstartrow = tickerNewRow - currentTickerStart 'use this calc to try and pinpoint where starting row for this ticker is

' Calculate Quarterly Change and Populate Column J

Closing\_Price = ws.Cells(i - 1, "F").Value

Opening\_Price = ws.Cells(currenttickerstartrow, "C").Value

Qtly\_Change = Closing\_Price - Opening\_Price

ws.Cells(tickerNewRow, "J").Value = Qtly\_Change

End If

Next i

Next ws

End Sub

***Error of mismatched type pops up somewhere new***

**debug type mismatch in Closing\_Price = ws.Cells(i - 1, "F").Value for following code: Sub stock\_alphabet() Dim Ticker As Variant Dim Qtly\_Change As Double Dim Opening\_Price As Double Dim Closing\_Price As Double Dim Percent\_Change As Variant Dim Total\_Stock\_Vol As Long Dim LastRow As Long Dim ws As Worksheet Dim currentTickerStart As Long Dim currentTickerValue As Variant Dim newtickerstart As Long Dim currenttickerstartrow As Long For Each ws In Worksheets ' Add column titles ws.Cells(1, 9).Value = "Ticker" ws.Cells(1, 10).Value = "Quarterly Change" ws.Cells(1, 11).Value = "Percent Change" ws.Cells(1, 12).Value = "Total Stock Volume" ' Find the last row for each worksheet LastRow = ws.Cells(ws.Rows.count, 1).End(xlUp).Row tickerNewRow = 2 currentTickerStart = 0 ' Copy the first ticker value to the new column ws.Cells(tickerNewRow, "I").Value = ws.Cells(2, "A").Value ' Loop through rows starting from the second row. define more variables will need For i = 2 To LastRow If ws.Cells(i, 1).Value <> ws.Cells(i - 1, 1).Value Then tickerNewRow = tickerNewRow + 1 ws.Cells(tickerNewRow, "I").Value = ws.Cells(i, "A").Value currentTickerValue = ws.Cells(i, "A").Value currentTickerStart = currentTickerStart + 1 'add 1 for each loop completed of i until not meet condition currenttickerstartrow = tickerNewRow - currentTickerStart 'use this calc to try and pinpoint where starting row for this ticker is ' Calculate Quarterly Change and Populate Column J Closing\_Price = ws.Cells(i - 1, "F").Value Opening\_Price = ws.Cells(currenttickerstartrow, "C").Value Qtly\_Change = Closing\_Price - Opening\_Price ws.Cells(tickerNewRow, "J").Value = Qtly\_Change End If Next i Next ws End Sub**

To address the type mismatch error in the line Closing\_Price = ws.Cells(i - 1, "F").Value, you can make sure that the cell values are explicitly converted to Double data type before performing arithmetic operations. Also, it's essential to handle cases where the cell values might not be numeric.

Here's an updated section of your code with modifications to handle the type conversion and potential non-numeric values:

' Calculate Quarterly Change and Populate Column J

If IsNumeric(ws.Cells(i - 1, "F").Value) Then

Closing\_Price = CDbl(ws.Cells(i - 1, "F").Value)

Else

' Handle non-numeric or empty cell values

' You can set a default value or handle the error as needed

Closing\_Price = 0 ' Set a default value, adjust as necessary

End If

Opening\_Price = CDbl(ws.Cells(currenttickerstartrow, "C").Value)

' Check if both Closing\_Price and Opening\_Price are numeric before calculation

If IsNumeric(Closing\_Price) And IsNumeric(Opening\_Price) Then

Qtly\_Change = Closing\_Price - Opening\_Price

ws.Cells(tickerNewRow, "J").Value = Qtly\_Change

Else

' Handle the case where the values are not numeric

' You can add error handling or set a default value

ws.Cells(tickerNewRow, "J").Value = "N/A"

End If

By explicitly converting the cell values to Double, checking for numeric values, and handling non-numeric cases, you can prevent type mismatch errors and ensure the calculation of Qtly\_Change proceeds correctly. Make sure to adjust the default values or error handling based on your specific requirements.

***Debug help bc populates but Qtly\_change values is off and first ticker value of repeats when populating in column J when supposed to appear once like other ticker values***

To address the issues you mentioned where the Qtly\_Change values are incorrect and the first ticker value repeats when populating in column J, you need to adjust the logic for calculating the quarterly change and ensure that the ticker values are populated correctly.

Here are some modifications you can make to your VBA script:

1. **Correct Quarterly Change Calculation**: Ensure that the quarterly change calculation is done correctly for each ticker. This calculation should be based on the opening and closing prices for each ticker.
2. **Populate Ticker Value Once**: Make sure that the ticker value is populated only once for each unique ticker in column I.
3. **Update TickerNewRow Appropriately**: Adjust the tickerNewRow variable to ensure that the correct row is updated when populating the ticker values and quarterly changes.

Here's an updated version of your script with these modifications:

Sub stock\_alphabet()

Dim Qtly\_Change As Double

Dim Opening\_Price As Double

Dim Closing\_Price As Double

Dim LastRow As Long

Dim ws As Worksheet

Dim tickerNewRow As Long

Dim currentTickerStart As Long

Dim currentTickerValue As Variant

Dim currenttickerstartrow As Long

For Each ws In Worksheets

' Add column titles

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quarterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

' Find the last row for each worksheet

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

tickerNewRow = 2

currentTickerStart = 0

' Loop through rows starting from the second row

For i = 2 To LastRow

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

tickerNewRow = tickerNewRow + 1

ws.Cells(tickerNewRow, "I").Value = ws.Cells(i, "A").Value

currentTickerValue = ws.Cells(i, "A").Value

currentTickerStart = 1 ' Reset the start for each new ticker

currenttickerstartrow = tickerNewRow

Else

' Calculate Quarterly Change and Populate Column J

If IsNumeric(ws.Cells(i, "F").Value) And IsNumeric(ws.Cells(i - 1, "C").Value) Then

Closing\_Price = CDbl(ws.Cells(i, "F").Value)

Opening\_Price = CDbl(ws.Cells(i - 1, "C").Value)

Qtly\_Change = Closing\_Price - Opening\_Price

ws.Cells(currenttickerstartrow, "J").Value = Qtly\_Change

Else

' Handle non-numeric values

ws.Cells(currenttickerstartrow, "J").Value = "N/A"

End If

End If

Next i

Next ws

End Sub

In this updated script:

* The tickerNewRow is adjusted to ensure the correct row is updated for each ticker.
* The quarterly change calculation is done based on the opening and closing prices for each ticker.
* The currentTickerStart is reset for each new ticker to correctly identify the starting row for each ticker.

***Worked on own plus above and got above. Logically seems to make sense to calc QtlyChang for AAB (1st ticker), but falls apart for next ticker bc startlocation comes up as 2 also. Since adding “i” to both counters that start at 0 and 2 respectively…***

Sub stock\_alphabet()

Dim Ticker As Variant 'do i even need this?

Dim Qtly\_Change As Double

Dim currentTickerOpen As Double

Dim currentTickerClose As Double

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Long

Dim LastRow As Long

Dim ws As Worksheet

Dim rowcounter As Long

'Dim currentTickerValue As Variant

Dim newtickerstart As Long

Dim currenttickerstartrow As Long

Dim currentrowlocation As Long

For Each ws In Worksheets

' Add column titles

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quarterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

' Find the last row for each worksheet

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

tickerNewRow = 1

rowcounter = 0

' Loop through rows starting from the second row

For i = 2 To LastRow

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

tickerNewRow = tickerNewRow + 1

ws.Cells(tickerNewRow, "I").Value = ws.Cells(i, "A").Value

'currentTickerValue = ws.Cells(i, "A").Value 'do i even need this?

' Calculate Quarterly Change and Populate Column J

rowcounter = rowcounter + i 'add 1 for each loop completed of i until not meet condition

currentrowlocation = 2 + i 'starting from 2, add 1 for each loop done until not meet condition

currenttickerstartrow = currentrowlocation - rowcounter 'use this calc to try and pinpoint where starting row for this ticker is

currentTickerOpen = ws.Cells(currenttickerstartrow, "C").Value

currentTickerClose = ws.Cells(i + 2, "F").Value

Qtly\_Change = currentTickerClose - currentTickerOpen

ws.Cells(tickerNewRow, "J").Value = Qtly\_Change

End If

Next i

Next ws

End Sub

***Populates ticker values correctly, but QtlyChange values still off and after a certain point they are the same value.***

**vba code for calculating amount of loops that meet the conditional between each unique ticker value not meeting conditional in same column. for example A2:A224 is "AAB" ticker value, A225:A507 is "AAF", and A508:A760 is "AAR" and so on. how to figure out the row where the condition is not met (the unique ticker value is now different) to make calculations based off this**

To calculate the number of loops that meet a condition between each unique ticker value and identify the row where the condition is not met (the unique ticker value changes), you can use VBA in Excel. Here's a general outline of how you can approach this:

1. **Loop Through the Data:**
   * Use a For loop to iterate through the rows in the column containing the ticker values.
   * Compare the current ticker value with the previous one to identify when the ticker value changes.
2. **Check Condition and Count Loops:**
   * Within the loop, check the condition you are interested in for each row.
   * If the condition is met, increment a counter variable to keep track of the number of loops that meet the condition between each unique ticker value.
3. **Identify Row Where Condition is Not Met:**
   * When the ticker value changes (i.e., the condition is not met for the current row), store the row number where this change occurs.
   * You can use a variable to keep track of the last row where the condition was met and update it when the condition is no longer met.

Here's a simplified example to illustrate this concept:

Sub CalculateLoops()

Dim lastTicker As String

Dim lastRowWithCondition As Long

Dim loopCount As Long

lastTicker = Cells(2, 1).Value ' Assuming ticker values start from row 2 in column A

lastRowWithCondition = 2

loopCount = 0

For i = 3 To 1000 ' Assuming data goes up to row 1000

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

' Condition not met, calculate based on previous condition

' Example: Calculate the sum of values between lastRowWithCondition and i-1

Dim sum As Double

For j = lastRowWithCondition To i - 1

sum = sum + Cells(j, 2).Value ' Assuming values are in column B

Next j

Debug.Print "Sum between " & lastRowWithCondition & " and " & i - 1 & ": " & sum

' Update lastRowWithCondition and lastTicker

lastRowWithCondition = i

lastTicker = Cells(i, 1).Value

loopCount = loopCount + 1

End If

Next i

' Final calculations for the last set of rows

Dim finalSum As Double

For k = lastRowWithCondition To 1000

finalSum = finalSum + Cells(k, 2).Value

Next k

Debug.Print "Final Sum from " & lastRowWithCondition & " to 1000: " & finalSum

Debug.Print "Number of loops meeting condition: " & loopCount

End Sub

Please adapt this code to suit your specific data structure and condition for the calculation. This is a basic example to get you started on calculating loops meeting a condition between each unique ticker value and identifying the row where the condition is not met.

***Tried to add adjustments below***

Sub stock\_alphabet()

Dim Ticker As Variant 'do i even need this?

Dim Qtly\_Change As Double

Dim currentTickerOpen As Double

Dim currentTickerClose As Double

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Long

Dim LastRow As Long

Dim ws As Worksheet

Dim loopcounter As Long

'Dim currentTickerValue As Variant 'not sure I need this

'Dim newtickerstart As Long

'Dim currenttickerstartrow As Long

'Dim currentrowlocation As Long

For Each ws In Worksheets

' Add column titles

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quarterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

' Find the last row for each worksheet

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

tickerNewRow = 1

' Loop through rows starting from the second row

For i = 2 To LastRow

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

tickerNewRow = tickerNewRow + 1

ws.Cells(tickerNewRow, "I").Value = ws.Cells(i, "A").Value

'currentTickerValue = ws.Cells(i, "A").Value 'do i even need this?

Dim lastTicker As String

Dim lastRowWithCondition As Long

Dim loopCount As Long

lastTicker = Cells(2, 1).Value ' Assuming ticker values start from row 2 in column A

lastRowWithCondition = 2

loopCount = 0

End If

For j = 3 To LastRow

If Cells(j, 1).Value <> lastTicker Then

' Condition not met, calculate based on previous condition

' Example: Calculate the sum of values between lastRowWithCondition and j-1

Dim sum As Double

For j = lastRowWithCondition To j - 1

'sum = sum + Cells(j, 2).Value ' Assuming values are in column B

sum = sum + j

Next j

Debug.Print "Sum between " & lastRowWithCondition & " and " & j - 1 & ": " & sum

' Update lastRowWithCondition and lastTicker

lastRowWithCondition = j

lastTicker = Cells(j, 1).Value

loopCount = loopCount + 1

End If

Next j

' Final calculations for the last set of rows

Dim finalSum As Double

For k = lastRowWithCondition To LastRow

finalSum = finalSum + Cells(k, 2).Value

Next k

' Calculate Quarterly Change and Populate Column J

'loopcounter = (-2)

'loopcounter = loopcounter + i 'add 1 for each loop completed of i until not meet condition

'currentrowlocation = i 'starting from 2, add 1 for each loop done until not meet condition. that's the current row

'currenttickerstartrow = currentrowlocation - [] 'use this calc to try and pinpoint where starting row for this ticker is

'currentTickerOpen = ws.Cells(currenttickerstartrow, "C").Value

'currentTickerClose = ws.Cells(i + 2, "F").Value

'Qtly\_Change = currentTickerClose - currentTickerOpen

'ws.Cells(tickerNewRow, "J").Value = Qtly\_Change

Next i

Next ws

End Sub

***Played around with it some, trying to fix the logic, consolidate etc***

Sub stock\_alphabet()

Dim Ticker As Variant 'do i even need this?

Dim Qtly\_Change As Double

Dim currentTickerOpen As Double

Dim currentTickerClose As Double

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Long

Dim LastRow As Long

Dim ws As Worksheet

Dim loopcounter As Long

'Dim currentTickerValue As Variant 'not sure I need this

'Dim newtickerstart As Long

Dim currenttickerstartrow As Long

'Dim currentrowlocation As Long

For Each ws In Worksheets

' Add column titles

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quarterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

' Find the last row for each worksheet

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

tickerNewRow = 1

' Loop through rows starting from the second row. if condition not met, start new row in column I that has new ticker ID

For i = 2 To LastRow

If ws.Cells(i, "A").Value <> ws.Cells(i - 1, "A").Value Then

tickerNewRow = tickerNewRow + 1

ws.Cells(tickerNewRow, "I").Value = ws.Cells(i, "A").Value

Dim newTickerValue As String

Dim lastRowWithCondition As Long

'Dim loopCount As Long

'update current ticker value showing to be what is in the condition breaking row (i)

newTickerValue = ws.Cells(i, "A").Value

lastRowWithCondition = i - 1 'row before condition breaks (condition breaks in i)

loopCount = 0

End If

For j = lastRowWithCondition To 1 Step -1

'if ticker value not the same as ticker above it

If Cells(j, "A").Value <> Cells(j - 1, "A").Value Then

' Condition not met, calculate based on previous condition

Dim rowdifference As Double

Dim endcell As Range

'For k = lastRowWithCondition To j - 1

'set start row for ticker

currenttickerstartrow = j

currentTickerOpen = Cells(currenttickerstartrow, "C").Value

currentTickerClose = Cells(lastRowWithCondition, "F").Value

Qtly\_Change = currentTickerClose - currentTickerOpen

ws.Cells(tickerNewRow, "J").Value = Qtly\_Change

'Set currentTickerOpen = ws.Cells(j, 3).Value ' opening price for

'set end cell

'Set endcell = currenttickerstartrow.End(xlDown)

'rowdifference = endcell.Row - currenttickerstartrow.Row + 1 'can this be used to get to my open price?

' Next k

' Update lastRowWithCondition and lastTicker

'lastRowWithCondition = j

'currentTickerValue = Cells(j, 1).Value

'loopCount = loopCount + 1

End If

Next j

' Final calculations for the last set of rows

'Dim finalSum As Double

'For h = lastRowWithCondition To LastRow

'finalSum = finalSum + Cells(h, 2).Value

'Next h

' Calculate Quarterly Change and Populate Column J

'loopcounter = (-2)

'loopcounter = loopcounter + i 'add 1 for each loop completed of i until not meet condition

'currentrowlocation = i 'starting from 2, add 1 for each loop done until not meet condition. that's the current row

'currenttickerstartrow = currentrowlocation - [] 'use this calc to try and pinpoint where starting row for this ticker is

'currentTickerOpen = ws.Cells(currenttickerstartrow, "C").Value

'currentTickerClose = ws.Cells(i + 2, "F").Value

'Qtly\_Change = currentTickerClose - currentTickerOpen

'ws.Cells(tickerNewRow, "J").Value = Qtly\_Change

Next i

Next ws

End Sub

***Error in defining object or application***

It seems like you are trying to create a VBA script to loop through stock market records and calculate quarterly changes based on certain conditions.

The error you mentioned, "If ws.Cells(j, "A").Value <> ws.Cells(j - 1, "A").Value Then", typically occurs when there is an issue with the loop boundaries or the logic within the loop.

In your code, the loop that is causing the issue is:

For j = lastRowWithCondition To 1 Step -1

If ws.Cells(j, "A").Value <> ws.Cells(j - 1, "A").Value Then

' Calculate Quarterly Change based on the condition

currenttickerstartrow = j

currentTickerOpen = ws.Cells(currenttickerstartrow, "C").Value

currentTickerClose = ws.Cells(lastRowWithCondition, "F").Value

Qtly\_Change = currentTickerClose - currentTickerOpen

ws.Cells(tickerNewRow, "J").Value = Qtly\_Change

End If

Next j

**The issue might be related to how you are handling the loop boundaries or the logic within the loop. Make sure that the loop is correctly iterating over the rows and that the conditions are properly evaluated.**

**I recommend reviewing the logic within this loop, especially how you are checking for the change in ticker symbol and calculating the quarterly change. You may need to adjust the loop boundaries or the conditions to ensure it behaves as expected.**

If you encounter any specific errors or need further assistance, please let me know!

***Populates but not J2 and wrong values for qtly change***

Sub stock\_alphabet()

Dim Ticker As Variant 'do i even need this?

Dim Qtly\_Change As Variant

Dim currentTickerOpen As Variant

Dim currentTickerClose As Variant

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Variant

Dim LastRow As Variant

Dim ws As Worksheet

Dim newTickerValue As Variant

Dim lastRowWithCondition As Variant

Dim currenttickerstartrow As Variant

'Dim currentTickerValue As Variant 'not sure I need this

'Dim newtickerstart As Long

'Dim currentrowlocation As Long

For Each ws In Worksheets

' Add column titles

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quarterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

' Find the last row for each worksheet

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

tickerNewRow = 1

' Loop through rows starting from the second row. if condition not met, start new row in column I that has new ticker ID

For i = 2 To LastRow

If ws.Cells(i, "A").Value <> ws.Cells(i - 1, "A").Value Then

tickerNewRow = tickerNewRow + 1

ws.Cells(tickerNewRow, "I").Value = ws.Cells(i, "A").Value

'Dim loopCount As Long

'update current ticker value showing to be what is in the condition breaking row (i)

newTickerValue = ws.Cells(i, "A").Value

lastRowWithCondition = i - 1 'row before condition breaks (condition breaks in i)

For j = lastRowWithCondition To 2 Step -1

'if value w/in LadtRowWithCondition is not same as one above it

If ws.Cells(j, "A").Value <> ws.Cells(j - 1, "A").Value Then

' Calculate Quarterly Change based on the condition

currenttickerstartrow = j

currentTickerOpen = ws.Cells(currenttickerstartrow, "C").Value

currentTickerClose = ws.Cells(lastRowWithCondition, "F").Value

Qtly\_Change = currentTickerClose - currentTickerOpen

ws.Cells(tickerNewRow, "J").Value = Qtly\_Change

End If

Next j

'Set currentTickerOpen = ws.Cells(j, 3).Value ' opening price for

'set end cell

'Set endcell = currenttickerstartrow.End(xlDown)

'rowdifference = endcell.Row - currenttickerstartrow.Row + 1 'can this be used to get to my open price?

' Next k

' Update lastRowWithCondition and lastTicker

'lastRowWithCondition = j

'currentTickerValue = Cells(j, 1).Value

'loopCount = loopCount + 1

' Final calculations for the last set of rows

'Dim finalSum As Double

'For h = lastRowWithCondition To LastRow

'finalSum = finalSum + Cells(h, 2).Value

'Next h

' Calculate Quarterly Change and Populate Column J

'loopcounter = (-2)

'loopcounter = loopcounter + i 'add 1 for each loop completed of i until not meet condition

'currentrowlocation = i 'starting from 2, add 1 for each loop done until not meet condition. that's the current row

'currenttickerstartrow = currentrowlocation - [] 'use this calc to try and pinpoint where starting row for this ticker is

'currentTickerOpen = ws.Cells(currenttickerstartrow, "C").Value

'currentTickerClose = ws.Cells(i + 2, "F").Value

'Qtly\_Change = currentTickerClose - currentTickerOpen

'ws.Cells(tickerNewRow, "J").Value = Qtly\_Change

End If

Next i

Next ws

End Sub

***Played around more to add percent change and total stock volume. Issue with my syntax in total stock volume***

Sub stock\_alphabet()

Dim Ticker As Variant 'do i even need this?

Dim Qtly\_Change As Variant

Dim currentTickerOpen As Variant

Dim currentTickerClose As Variant

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Variant

Dim LastRow As Variant

Dim ws As Worksheet

Dim newTickerValue As Variant

Dim lastRowWithCondition As Variant

Dim currenttickerstartrow As Variant

'Dim FirstQty\_Change As Variant

'Dim Firsttickeropen As Variant

'Dim firsttickerclose As Variant

'Dim currentTickerValue As Variant 'not sure I need this

'Dim newtickerstart As Long

'Dim currentrowlocation As Long

For Each ws In Worksheets

' Add column titles

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quarterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

' Find the last row for each worksheet

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

tickerNewRow = 1

' Loop through rows starting from the second row. if condition not met, start new row in column I that has new ticker ID

For i = 2 To LastRow

If ws.Cells(i, "A").Value <> ws.Cells(i - 1, "A").Value Then

tickerNewRow = tickerNewRow + 1

ws.Cells(tickerNewRow, "I").Value = ws.Cells(i, "A").Value

'Dim loopCount As Long

'update current ticker value showing to be what is in the condition breaking row (i)

newTickerValue = ws.Cells(i, "A").Value

lastRowWithCondition = i - 1 'row before condition breaks (condition breaks in i)

For j = lastRowWithCondition To 2 Step -1

'if value w/in LadtRowWithCondition is not same as one above it

If ws.Cells(j, "A").Value <> ws.Cells(j - 1, "A").Value Then

' Calculate Quarterly Change based on the condition

currenttickerstartrow = j

currentTickerOpen = ws.Cells(currenttickerstartrow, "C").Value

currentTickerClose = ws.Cells(lastRowWithCondition, "F").Value

Qtly\_Change = currentTickerClose - currentTickerOpen

ws.Cells(tickerNewRow, "J").Value = Qtly\_Change

'do color code for qtly\_change value dep if over(green), red(under), or same as 0(blank)

'calc percent change and populate it

Percent\_Change = (Qtly\_Change / currentTickerOpen)

ws.Cells(tickerNewRow, "K").Value = Percent\_Change

'calc total stock vol for quarter for each ticker. populate it

Total\_Stock\_Vol = WorksheetFunction.sum("G"&"currenttickerstartrow":"G"&"lastRowWithCondition")

End If

Next j

' Final calculations for the last set of rows

'For h = lastRowWithCondition To LastRow

'Firsttickeropen = ws.Cells(2, "C").Value

'firsttickerclose = ws.Cells((i - 1), "F").Value

'[] -[] = FirstQty\_Change

'FirstQty\_Change = ws.Cells().Value

'Next h

'Set currentTickerOpen = ws.Cells(j, 3).Value ' opening price for

'set end cell

'Set endcell = currenttickerstartrow.End(xlDown)

'rowdifference = endcell.Row - currenttickerstartrow.Row + 1 'can this be used to get to my open price?

' Next k

' Update lastRowWithCondition and lastTicker

'lastRowWithCondition = j

'currentTickerValue = Cells(j, 1).Value

'loopCount = loopCount + 1

' Calculate Quarterly Change and Populate Column J

'loopcounter = (-2)

'loopcounter = loopcounter + i 'add 1 for each loop completed of i until not meet condition

'currentrowlocation = i 'starting from 2, add 1 for each loop done until not meet condition. that's the current row

'currenttickerstartrow = currentrowlocation - [] 'use this calc to try and pinpoint where starting row for this ticker is

'currentTickerOpen = ws.Cells(currenttickerstartrow, "C").Value

'currentTickerClose = ws.Cells(i + 2, "F").Value

'Qtly\_Change = currentTickerClose - currentTickerOpen

'ws.Cells(tickerNewRow, "J").Value = Qtly\_Change

End If

Next i

Next ws

End Sub

In your code snippet, the line where you calculate the Total\_Stock\_Vol seems to have a syntax issue. When concatenating the cell references to form the range for the WorksheetFunction.Sum method, you need to ensure that the cell references are concatenated properly.

Here is the corrected line for calculating Total\_Stock\_Vol:

Total\_Stock\_Vol = WorksheetFunction.Sum(ws.Range("G" & currenttickerstartrow & ":G" & lastRowWithCondition))

In this corrected line:

* ws.Range("G" & currenttickerstartrow & ":G" & lastRowWithCondition) forms the range from cell G(currenttickerstartrow) to G(lastRowWithCondition) on the current worksheet ws.
* WorksheetFunction.Sum() is then used to calculate the sum of the values in that range.

Make sure to replace the line in your code with this corrected line to calculate the total stock volume correctly.

**vba script for highest value in column and for lowest value in column**

To find the highest and lowest values in a specific column using VBA in Excel, you can use the following script as an example:

Sub FindHighestAndLowestValue()

Dim ws As Worksheet

Dim lastRow As Long

Dim col As Integer

Dim highestValue As Double

Dim lowestValue As Double

' Set the worksheet

Set ws = ThisWorkbook.Sheets("Sheet1")

' Set the column to search for highest and lowest values

col = 1 ' Column A

' Find the last row in the column

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

' Initialize highest and lowest values

highestValue = ws.Cells(1, col).Value

lowestValue = ws.Cells(1, col).Value

' Loop through the column to find the highest and lowest values

For i = 2 To lastRow

If ws.Cells(i, col).Value > highestValue Then

highestValue = ws.Cells(i, col).Value

End If

If ws.Cells(i, col).Value < lowestValue Then

lowestValue = ws.Cells(i, col).Value

End If

Next i

' Display the highest and lowest values

MsgBox "Highest Value in Column " & col & ": " & highestValue & vbCrLf & \_

"Lowest Value in Column " & col & ": " & lowestValue

End Sub

In this script:

* Change the worksheet name "Sheet1" to the name of the sheet where you want to find the highest and lowest values.
* Set the col variable to the column number you want to search for the highest and lowest values.
* The script will find the highest and lowest values in the specified column and display them in a message box.

Start code for

Dim rng as variant

Dim minVal as variant

Dim maxVal as variant

Dim rngVol as variant

Dim maxValVol as variant

‘add titles

Ws.cells(“N2”).Value = “Greatest Percent Increase”

Ws.cells(“N3”).Value = “Greatest Percent Decrease”

Ws.cells(“N4”).Value = “Greatest Total Volume”

Ws.cells(“O1”).Value=”Ticker”

Ws.cells(“O2”).Value=”Value”

‘calculate Greatest Percent Decrease

' Set the range to the column where you want to find the lowest value

Set rng = ws.Range("K2:”K”&”LastRow")

' Initialize minVal with the first cell value in the range

minVal = rng.Cells(1).Value

' Loop through each cell in the range to find the lowest value

For Each cell In rng

If cell.Value < minVal Then

minVal = cell.Value

End If

Next cell

Ws.cells(“P2”).Value = minVal

‘calculate Greatest Percent Increase

' Initialize maxVal with the first cell value in the range

maxVal = rng.Cells(1).Value

' Loop through each cell in the range to find the highest value

For Each cell In rng

If cell.Value > maxVal Then

maxVal = cell.Value

End If

Next cell

Ws.cells(“P3”).Value = maxVal

‘calculate Greatest Total Volume

' Set the range to the column where you want to find the highest value of vol

Set rngVol = ws.Range("L2:”L”&”LastRow")

' Initialize maxValVol with the first cell value in the range

maxValVol = rngVol.Cells(1).Value

' Loop through each cell in the range to find the lowest value

For Each cell In rngVol

If cell.Value < maxValVol Then

maxValVol = cell.Value

End If

Next cell

Ws.cells(“P4”).Value = minVal

Sub stock\_alphabet()

Dim Ticker As Variant 'do i even need this?

Dim Qtly\_Change As Variant

Dim currentTickerOpen As Variant

Dim currentTickerClose As Variant

Dim Percent\_Change As Variant

Dim Total\_Stock\_Vol As Variant

Dim LastRow As Variant

Dim ws As Worksheet

Dim newTickerValue As Variant

Dim lastRowWithCondition As Variant

Dim currenttickerstartrow As Variant

Dim rng As Variant

Dim minVal As Variant

Dim maxVal As Variant

Dim rngVol As Variant

Dim maxValVol As Variant

'Dim FirstQty\_Change As Variant

'Dim Firsttickeropen As Variant

'Dim firsttickerclose As Variant

'Dim currentTickerValue As Variant 'not sure I need this

'Dim newtickerstart As Long

'Dim currentrowlocation As Long

For Each ws In Worksheets

' Add column titles

ws.Cells(1, 9).Value = "Ticker"

ws.Cells(1, 10).Value = "Quarterly Change"

ws.Cells(1, 11).Value = "Percent Change"

ws.Cells(1, 12).Value = "Total Stock Volume"

' Find the last row for each worksheet

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

tickerNewRow = 1

' Loop through rows starting from the second row. if condition not met, start new row in column I that has new ticker ID

For i = 2 To LastRow

If ws.Cells(i, "A").Value <> ws.Cells(i - 1, "A").Value Then

tickerNewRow = tickerNewRow + 1

ws.Cells(tickerNewRow, "I").Value = ws.Cells(i, "A").Value

'Dim loopCount As Long

'update current ticker value showing to be what is in the condition breaking row (i)

newTickerValue = ws.Cells(i, "A").Value

lastRowWithCondition = i - 1 'row before condition breaks (condition breaks in i)

For j = lastRowWithCondition To 2 Step -1

'if value w/in LadtRowWithCondition is not same as one above it

If ws.Cells(j, "A").Value <> ws.Cells(j - 1, "A").Value Then

' Calculate Quarterly Change based on the condition

currenttickerstartrow = j

currentTickerOpen = ws.Cells(currenttickerstartrow, "C").Value

currentTickerClose = ws.Cells(lastRowWithCondition, "F").Value

Qtly\_Change = currentTickerClose - currentTickerOpen

ws.Cells(tickerNewRow, "J").Value = Qtly\_Change

If Qtly\_Change > 0 Then

ws.Cells(tickerNewRow, "J").Interior.Color = RGB(0, 255, 0)

ElseIf Qtly\_Change < 0 Then

ws.Cells(tickerNewRow, "J").Interior.Color = RGB(255, 0, 0)

Else: ws.Cells(tickerNewRow, "J").Interior.Color = xlNone

End If

'do color code for qtly\_change value dep if over(green), red(under), or same as 0(blank)

'calc percent change and populate it

Percent\_Change = (Qtly\_Change / currentTickerOpen)

ws.Cells(tickerNewRow, "K").Value = Percent\_Change

'calc total stock vol for quarter for each ticker. populate it

Total\_Stock\_Vol = WorksheetFunction.sum(ws.Range("G" & currenttickerstartrow & ":G" & lastRowWithCondition))

ws.Cells(tickerNewRow, "L").Value = Total\_Stock\_Vol

End If

Next j

'add titles

ws.Cells(2, "N").Value = "Greatest Percent Increase"

ws.Cells(3, "N").Value = "Greatest Percent Decrease"

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

ws.Cells(1, "O").Value = "Ticker"

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

'calculate Greatest Percent Decrease

' Set the range to the column where you want to find the lowest value

Set rng = ws.Range("K2:K" & LastRow)

' Initialize minVal with the first cell value in the range

minVal = rng.Cells(1).Value

' Loop through each cell in the range to find the lowest value

For Each cell In rng

If cell.Value < minVal Then

minVal = cell.Value

End If

Next cell

ws.Cells(2, "P").Value = minVal

'calculate Greatest Percent Increase

' Initialize maxVal with the first cell value in the range

maxVal = rng.Cells(1).Value

' Loop through each cell in the range to find the highest value

For Each cell In rng

If cell.Value > maxVal Then

maxVal = cell.Value

End If

Next cell

ws.Cells(3, "P").Value = maxVal

'calculate Greatest Total Volume

' Set the range to the column where you want to find the highest value of vol

Set rngVol = ws.Range("L2:L" & LastRow)

' Initialize maxValVol with the first cell value in the range

maxValVol = rngVol.Cells(1).Value

' Loop through each cell in the range to find the lowest value

For Each cell In rngVol

If cell.Value < maxValVol Then

maxValVol = cell.Value

End If

Next cell

ws.Cells(4, "P").Value = minVal

' Final calculations for the last set of rows

'For h = lastRowWithCondition To LastRow

'Firsttickeropen = ws.Cells(2, "C").Value

'firsttickerclose = ws.Cells((i - 1), "F").Value

'[] -[] = FirstQty\_Change

'FirstQty\_Change = ws.Cells().Value

'Next h

'Set currentTickerOpen = ws.Cells(j, 3).Value ' opening price for

'set end cell

'Set endcell = currenttickerstartrow.End(xlDown)

'rowdifference = endcell.Row - currenttickerstartrow.Row + 1 'can this be used to get to my open price?

' Next k

' Update lastRowWithCondition and lastTicker

'lastRowWithCondition = j

'currentTickerValue = Cells(j, 1).Value

'loopCount = loopCount + 1

' Calculate Quarterly Change and Populate Column J

'loopcounter = (-2)

'loopcounter = loopcounter + i 'add 1 for each loop completed of i until not meet condition

'currentrowlocation = i 'starting from 2, add 1 for each loop done until not meet condition. that's the current row

'currenttickerstartrow = currentrowlocation - [] 'use this calc to try and pinpoint where starting row for this ticker is

'currentTickerOpen = ws.Cells(currenttickerstartrow, "C").Value

'currentTickerClose = ws.Cells(i + 2, "F").Value

'Qtly\_Change = currentTickerClose - currentTickerOpen

'ws.Cells(tickerNewRow, "J").Value = Qtly\_Change

End If

Next i

Next ws

End Sub