

# IE 5390 – Assignment - 03

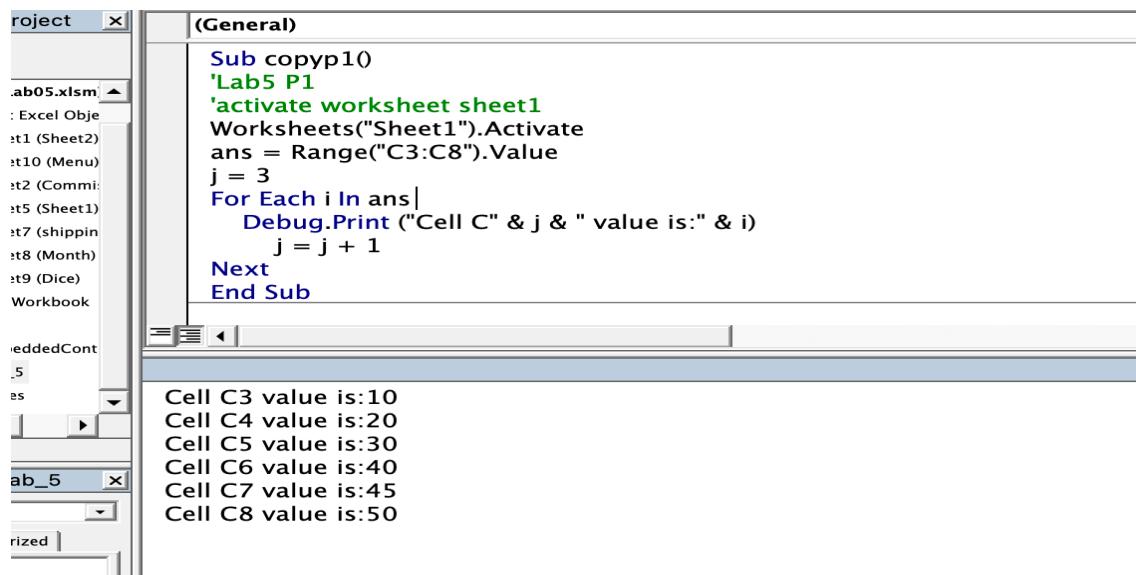
Name: Simran Abhay Sinha

## Lab 05

### 1. Using Sheet1 sheet

A. Write VBA code to copy values of cells C3:C8 into an array, then steps through the array and prints out each value

Code:



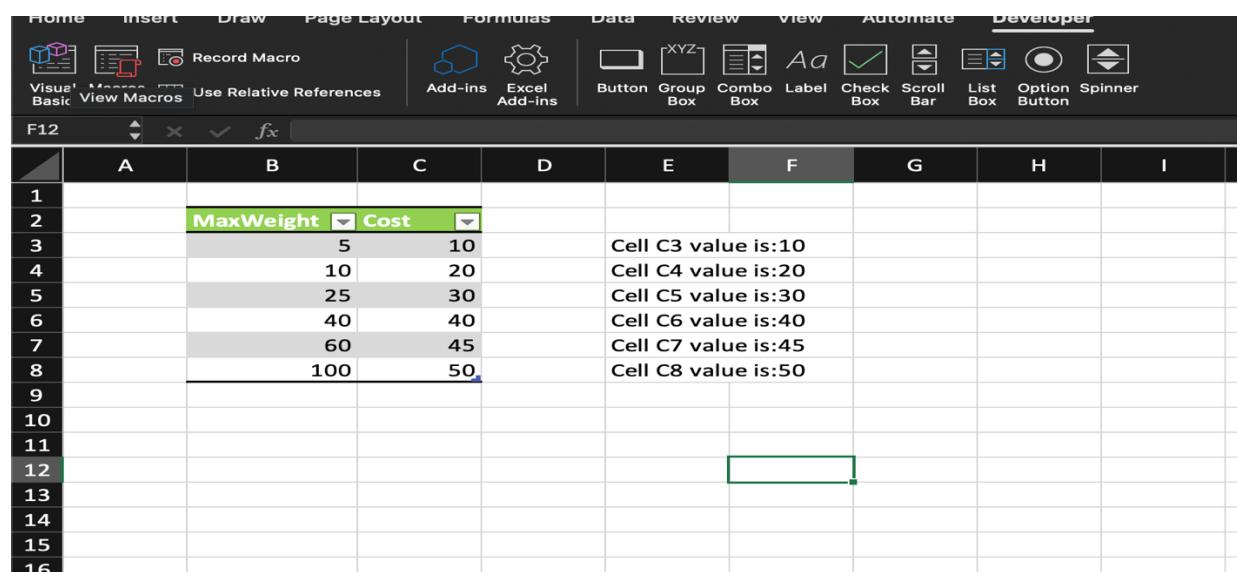
```

Sub copyp10
'Lab5 P1
'activate worksheet sheet1
Worksheets("Sheet1").Activate
ans = Range("C3:C8").Value
j = 3
For Each i In ans
    Debug.Print ("Cell C" & j & " value is:" & i)
    j = j + 1
Next
End Sub

```

The screenshot shows the Microsoft Visual Basic Editor (VBE) interface. The project tree on the left shows a single file named 'ab05.xlsm'. The code editor window contains the provided VBA code. The immediate window at the bottom displays the output of the code: 'Cell C3 value is:10', 'Cell C4 value is:20', 'Cell C5 value is:30', 'Cell C6 value is:40', 'Cell C7 value is:45', and 'Cell C8 value is:50'.

Result:



	A	B	C	D	E	F	G	H	I
1									
2		MaxWeight	Cost						
3		5	10			Cell C3 value is:10			
4		10	20			Cell C4 value is:20			
5		25	30			Cell C5 value is:30			
6		40	40			Cell C6 value is:40			
7		60	45			Cell C7 value is:45			
8		100	50			Cell C8 value is:50			
9									
10									
11									
12									
13									
14									
15									
16									

The screenshot shows an Excel spreadsheet with data in columns A and B, and the results of the VBA code execution in column F. The data in columns A and B is as follows:

	A	B
2	MaxWeight	Cost
3	5	10
4	10	20
5	25	30
6	40	40
7	60	45
8	100	50

The results in column F are:

- Cell C3 value is:10
- Cell C4 value is:20
- Cell C5 value is:30
- Cell C6 value is:40
- Cell C7 value is:45
- Cell C8 value is:50

## 2. Using Sheet2 sheet

## A. Calculate the simple average forecast of the time series

Code:

```

Sub saverage()
'Lab 5 P2
'calculate forecast
n = 10
Sum = 0
Worksheets("sheet2").Activate
For i = 1 To n
    Sum = Sum + Range("B" & i + 1)
Next i
Forecast = Sum / n
Range("D1").Value = "Simple Forecast is"
Range("D1").Columns.AutoFit
Range("D1").Interior.Color = RGB(0, 255, 255)
Range("D2").Value = Forecast
Range("D2").Interior.Color = RGB(0, 255, 255)
End Sub

```

Result:

	D	E
	Simple Forecast is	
	613.7	

B. Calculate the weighted average forecast of the time series. Use the following weights: {0.01, 0.02, 0.03, 0.04, 0.05, 0.1, 0.1, 0.15, 0.25, 0.25}. Note, the total sum of weights must equal to 1. We also put more weight on recent values than old ones as shown here b/c old values are OLD.

Code:

The screenshot shows the Microsoft Visual Basic Editor (VBE) interface. The menu bar includes File, Edit, View, Insert, Format, Debug, Run, Tools, and Window. The title bar reads "Microsoft Visual Basic - Lab05.xlsm - [Lab\_5 (Code)]". The status bar at the bottom right shows "Tue Jan 28 7:42PM".

The Project Explorer on the left lists "Project - VBAProject" with "Lab\_5" selected. It contains "References", "VBAProject (Lab06.xlsm)", and "Microsoft Excel Objects" which include "Sheet1 (CurrentStaff)", "Sheet2 (Phone Numbers)", "Sheet3 (ProductAttributes)", "Sheet4 (Sales)", "Sheet5 (Colors)", "Sheet6 (Products)", "Sheet7 (Products)", "Sheet8 (Capitalized)", and "ThisWorkbook". A "Modules" folder is also present.

The code editor window displays the following VBA code:

```
Sub weightedAverage()
'Lab 5 Problem 2 - B
Dim weightedSum As Double
Dim weights(1 To 10) As Double
Dim i As Integer

'Initialize variables
n = 10
weightedSum = 0

'Define weights (must sum to 1)
weights(1) = 0.01
weights(2) = 0.02
weights(3) = 0.03
weights(4) = 0.04
weights(5) = 0.05
weights(6) = 0.1
weights(7) = 0.1
weights(8) = 0.15
weights(9) = 0.25
weights(10) = 0.25
Worksheets("Sheet2").Activate

'Calculate weighted sum
For i = 1 To n
    weightedSum = weightedSum + (Range("B" & i + 1).Value * weights(i))
Next i

'Output results
Range("E1").Value = "Weighted Forecast is"
Range("E1").Columns.AutoFit
Range("E1").Interior.Color = RGB(0, 255, 255)
Range("E2").Value = weightedSum
Range("E2").Interior.Color = RGB(0, 255, 255)
```

### Result:

The screenshot shows a Microsoft Excel spreadsheet titled "Lab05". The ribbon menu is visible at the top, with the "Developer" tab selected. The formula bar shows "E11". The main content area contains a table with data and formulas.

	A	B	C	D	E	F	G	H	I	J	K	L
1	week	demand			Simple Forecast is	Weighted Forecast is						
2	1	500			613.7	652.65						
3	2	750										
4	3	300										
5	4	800										
6	5	400										
7	6	675										
8	7	823										
9	8	504										
10	9	586										
11	10	799										
12												
13												
14												
15												
16												
17												
18												
19												
20												
21												
22												
23												

At the bottom, there are tabs for "Sheet1", "Sheet2", "Commission", "shipping", "Month", "Dice", "Menu", and a "+" button. The status bar shows "Ready", "Accessibility: Investigate", and zoom levels from 189% to 100%.

C. calculate the minimum and maximum of the data

Code:

The screenshot shows the Microsoft Visual Basic Editor (VBE) interface. The menu bar includes File, Edit, View, Insert, Format, Debug, Run, Tools, and Window. The title bar reads "Microsoft Visual Basic - Lab05.xlsm - [Lab\_5 (Code)]". The status bar at the bottom right shows the date and time: "Tue Jan 28 7:45 PM".

The left pane displays the "Project - VBAProject" window for the "Lab\_5" project. It lists the following items:

- References
- VBAProject (Lab05.xlsm)
- Microsoft Excel Objects
  - Sheet1 (CurrentStaff)
  - Sheet2 (Phone Numbers)
  - Sheet3 (Product Attributes)
  - Sheet4 (Names)
  - Sheet5 (Codes)
  - Sheet6 (Invent)
  - Sheet7 (Products)
  - Sheet8 (Capitalized)
- ThisWorkbook
- Modules
  - Lab\_5 Module

The main code editor window contains two subroutines: "Calculate weighted sum" and "minmaxValue".

```
Sub Calculate_weighted_sum()
    Dim weightedSum As Double
    Dim i As Integer
    Dim weights() As Double

    weights(9) = 0.25
    weights(10) = 0.25
    Worksheets("Sheet2").Activate

    'Calculate weighted sum
    For i = 1 To n
        weightedSum = weightedSum + (Range("B" & i + 1).Value * weights(i))
    Next i

    'Output results
    Range("E1").Value = "Weighted Forecast is"
    Range("E1").Columns.AutoFit
    Range("E1").Interior.Color = RGB(0, 255, 255)
    Range("E2").Value = weightedSum
    Range("E2").Interior.Color = RGB(0, 255, 255)
End Sub

Sub minmaxValue()
    'Lab 5 P2C
    'minmaxValue Macro
    Range("D4").Select
    ActiveCell.FormulaR1C1 = "Minimum Value"
    Range("E4").Select
    ActiveCell.FormulaR1C1 = "=MIN(R[-2]C[-3]:R[7]C[-3])"
    Range("D5").Select
    ActiveCell.FormulaR1C1 = "Maximum Value"
    Range("E5").Select
    ActiveCell.FormulaR1C1 = "=MAX(R[-3]C[-3]:R[6]C[-3])"
    Range("E6").Select
    End Sub
```

The "Properties" window on the left is open for the "Lab\_5" module, showing the "Alphabetic" tab selected. The "Name" dropdown is set to "Lab\_5".

## Result:

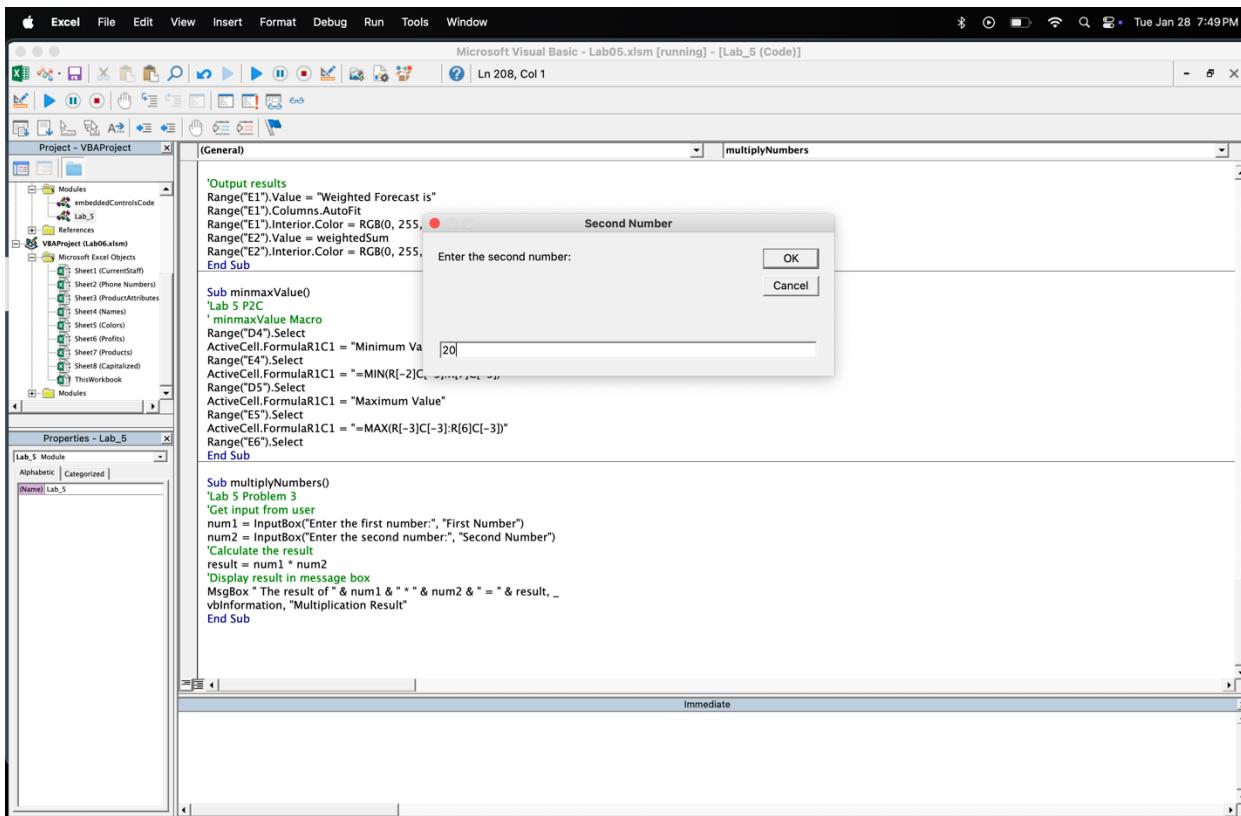
3. Write code that accepts two numbers and multiply them and display results in MsgBox  
Code:

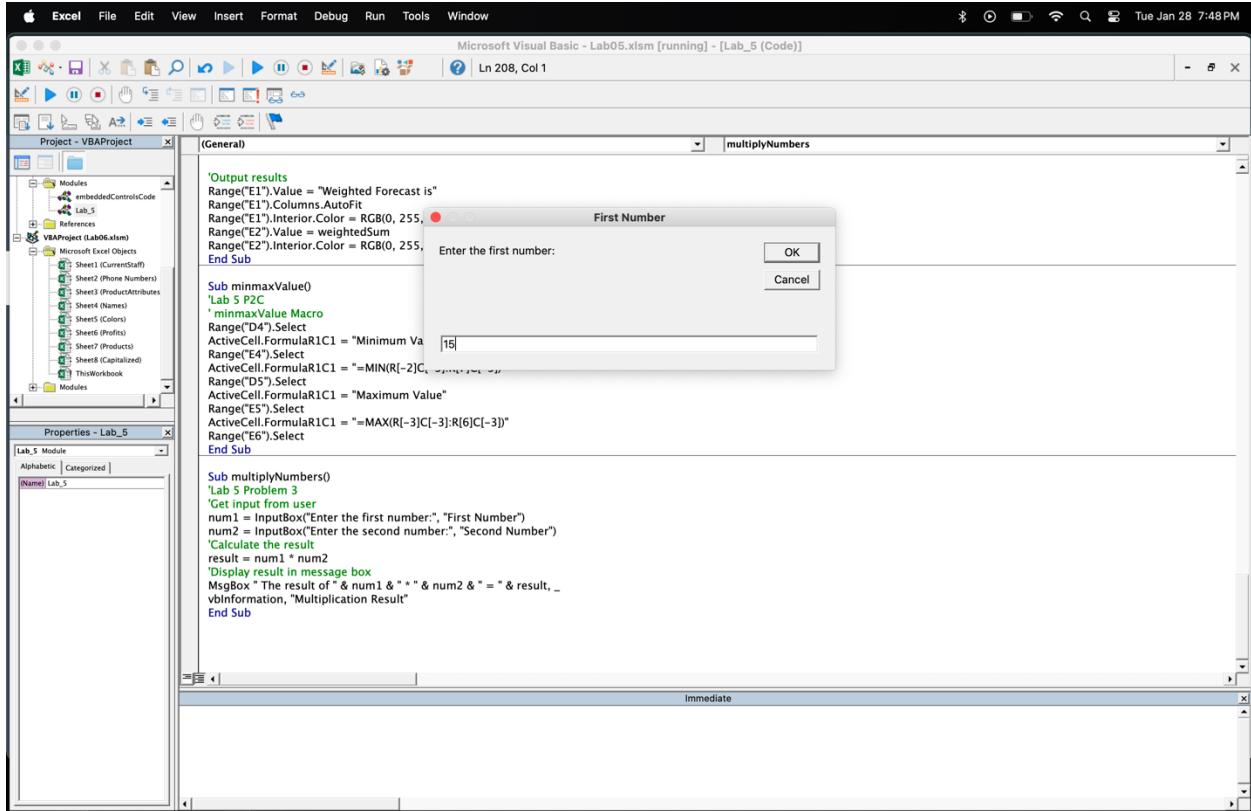
```

Lab_5 module
Alphabetic | Categorized | (Name) Lab_5
Sub multiplyNumbers()
    'Lab 5 Problem 3
    'Get input from user
    num1 = InputBox("Enter the first number.", "First Number")
    num2 = InputBox("Enter the second number.", "Second Number")
    'Calculate the result
    result = num1 * num2
    'Display result in message box
    MsgBox "The result of " & num1 & " * " & num2 & " = " & result, _
        vbInformation, "Multiplication Result"
End Sub

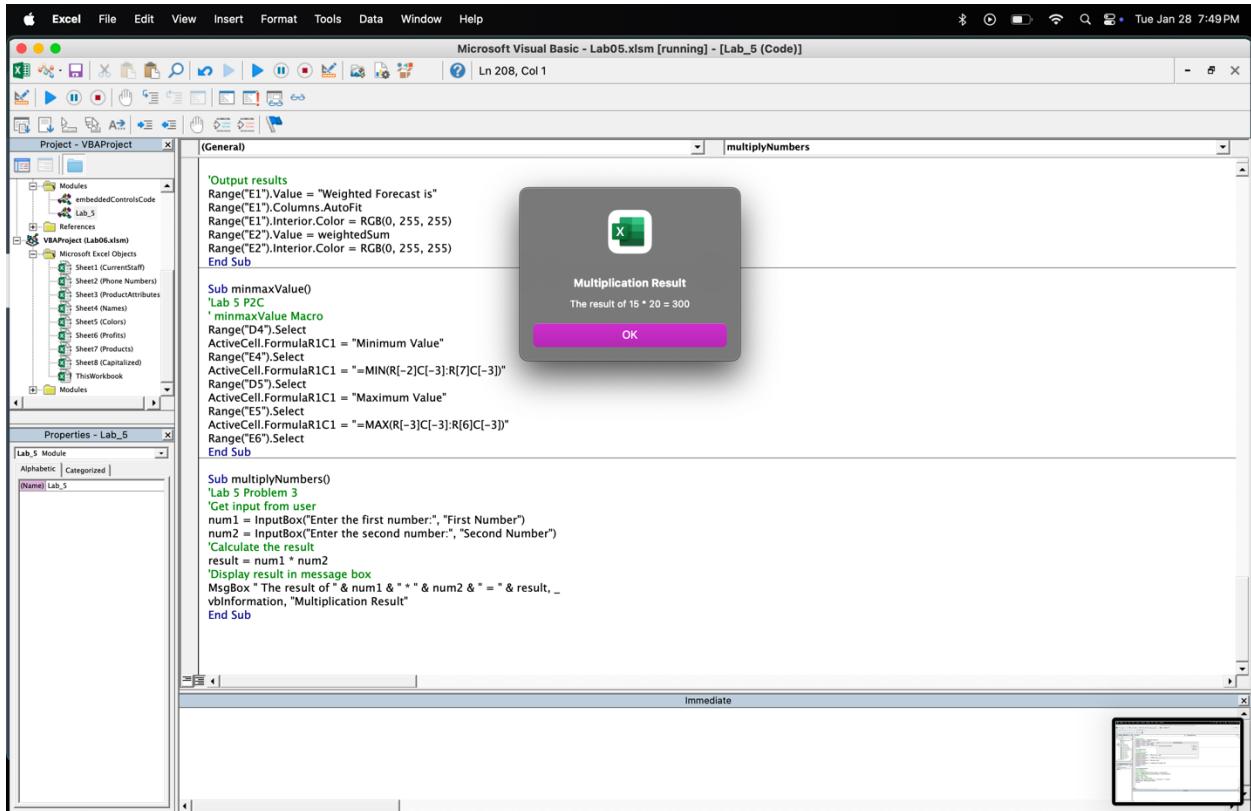
```

Input:





Result:



#### 4. Using Commission sheet

A. Write a subroutine that calculates a 15% commission of Unit Sales and displays it in column D

Code:

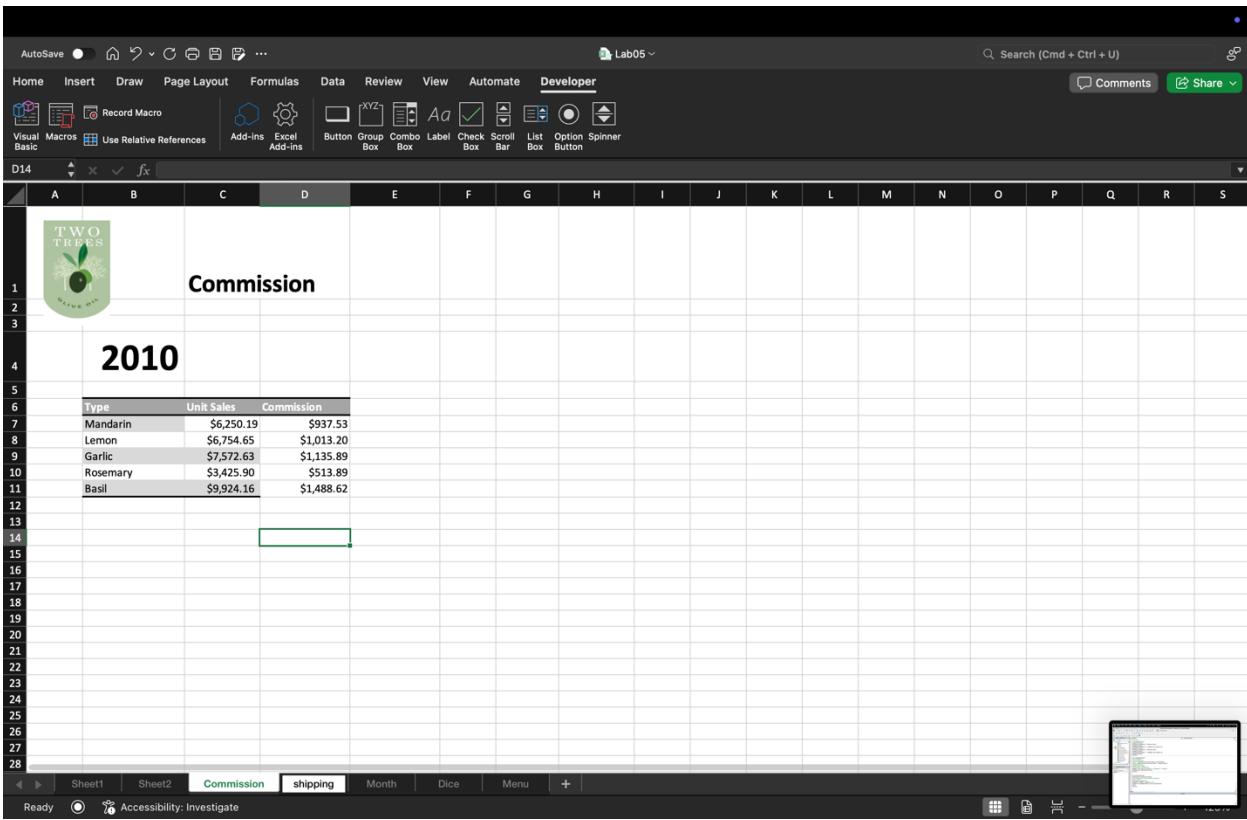


```

Sub commissionCalc()
    Worksheet("Commission").Activate
    'Loop through each row and calculate commission
    For i = 7 To 11
        salesValue = Range("C" & i)
        Range("D" & i).Value = salesValue * 0.15
        Range("D" & i).NumberFormat = "$-en-US#,##0.00"
    Next i
End Sub

```

Result:



The screenshot shows an Excel spreadsheet titled "Commission". The table contains the following data:

Type	Unit Sales	Commission
Mandarin	\$6,250.19	\$937.53
Lemon	\$6,754.65	\$1,013.20
Garlic	\$7,572.63	\$1,135.89
Rosemary	\$3,425.90	\$513.89
Basil	\$9,924.16	\$1,488.62

B. Repeat A but write a function that takes Unit Sales as input and displays commission in column D

Code:

```
Function calculateCommission(unitSales As Double) As Double
'Lab 5 Problem 4B
calculateCommission = unitSales * 0.15
End Function
```

Result:

The screenshot shows an Excel spreadsheet with the following data:

Type	Unit Sales	Commission
Mandarin	\$6,250.19	\$937.53
Lemon	\$6,754.65	\$1,013.20
Garlic	\$7,572.63	\$1,135.89
Rosemary	\$3,425.90	\$513.89
Basil	\$9,924.16	\$1,488.62

Type	Unit Sales	Commission
Mandarin	\$6,250.19	\$937.53
Lemon	\$6,754.65	\$1,013.20
Garlic	\$7,572.63	\$1,135.89
Rosemary	\$3,425.90	\$513.89
Basil	\$9,924.16	\$1,488.62

5. Using Shipping sheet, calculate the weighted average cost of the shipping cost

#### A. Display intermediate results in column D

Code:

```
Sub calculateIntermediateP50()
'Lab 5 P5A
Worksheets("Shipping").Activate
For i = 3 To 8
Range("D" & i).Value = Range("B" & i).Value * Range("C" & i).Value
Next i
End Sub
```

### Result:

The screenshot shows a Microsoft Excel spreadsheet titled "Lab05". The ribbon menu is visible at the top, with the "Developer" tab selected. The "Developer" tab includes icons for Visual Basic, Macros, Record Macro, Use Relative References, Add-ins, Excel Add-ins, and various form controls like Button, Group Box, Combo Box, Label, Check Box, Scroll Bar, List Box, Option Button, and Spinner.

The worksheet contains a data table with columns labeled "MaxWeight" and "Cost". The data is as follows:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA	
1		MaxWeight	Cost																									
2		5	10	50																								
3		50	50	2500																								
4		25	30	750																								
5		40	40	1600																								
6		60	45	2700																								
7		100	50	5000																								
8																												
9																												
10																												
11																												
12																												
13																												
14																												
15																												
16																												
17																												
18																												
19																												
20																												
21																												
22																												
23																												
24																												
25																												
26																												
27																												
28																												
29																												
30																												
31																												
32																												
33																												
34																												
35																												
36																												
37																												
38																												
39																												
40																												
41																												
42																												
43																												
44																												
45																												

The ribbon tabs include AutoSave, Home, Insert, Draw, Page Layout, Formulas, Review, View, Automate, Developer, and a search bar. The status bar at the bottom shows "Sheet1", "Sheet2", "Commission", "shipping", "Month", "Dice", "Menu", and "+". A context menu is open near cell F26, listing options like "Copy", "Cut", "Paste", "Format Cells", etc.

- B. Display this text in cell B10: "WACost"
- C. Display the result in cell C10
- D. Color both cells the same color as cells B2 and C2

Code:

```
Sub calculateWAFinal()
'Lab 5 P5 - B, C, D
Range("B10").Value = "WACost"
Range("C10").Value = Application.Sum(Range("D3:D8")) / Application.Sum(Range("B3:B8"))
' Match cell colors
Dim headerColor As Long
headerColor = Range("B2").Interior.Color
Range("B10,C10").Interior.Color = headerColor
End Sub|
```

Result:

MaxWeight	Cost
5	10
50	50
25	30
40	40
60	45
100	50
	5000

WACost	45
--------	----

6. Using VBA, change the Shipping Tab color to red and the Menu tab color to green

Code:

```
Sub ChangeTabColors()
'Lab 5 P6
Worksheets("Shipping").Tab.Color = RGB(255, 0, 0)
Worksheets("Menu").Tab.Color = RGB(0, 255, 0)
End Sub
```

## Result:

The screenshot shows a Microsoft Excel spreadsheet titled "Lab05". The ribbon menu is visible at the top, with the "Developer" tab selected. Under the "Developer" tab, there are buttons for "Visual Basic", "Macros", "Record Macro", "Add-ins", "Excel Add-ins", and "Use Relative References".

The worksheet contains a table with two columns: "MaxWeight" and "Cost". The data rows are as follows:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1																	
2																	
3		MaxWeight	Cost														
4		5	10														
5		10	20														
6		25	30														
7		40	40														
8		60	45														
		100	50														
9																	
10																	
11																	
12																	
13																	
14																	
15																	
16																	
17																	
18																	
19																	
20																	
21																	
22																	
23																	
24																	
25																	
26																	
27																	
28																	
29																	

The status bar at the bottom shows "Sheet1" is active, along with other tabs like "Sheet2", "Commission", "shipping", "Month", "Dice", "Menu", and a "+" button. The search bar at the top right says "Search (Cmd + Ctrl + U)".

## 7. Using Month sheet

A. Change cells in column A to “Times New Roman”, size 12, and red color

**Code:**

```
Sub formatMonthColumnA()
'Lab 5 P7A
With Worksheets("Month").Range("A3:A34")
    .Font.Name = "Times New Roman"
    .Font.Size = 12
    .Font.Color = RGB(255, 0, 0)
End With
End Sub
```

## Result:

AutoSave  Home Insert Draw Page Layout Formulas Data Review View Automate Developer

Record Macro  Use Relative References  Excel Add-ins  Aa

C8

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
1																
2																
3	Day	Sales														
4	1	1358														
5	2	623														
6	3	1170														
7	4	1103														
8	5	1376														
9	6	1304														
10	7	432														
11	8	133														
12	9	1100														
13	10	772														
14	11	1313														
15	12	1044														
16	13	289														
17	14	966														
18	15	1415														
19	16	1094														
20	17	884														
21	18	962														
22	19	1366														
23	20	677														
24	21	1141														
25	22	384														

Sheet1 Sheet2 Commission shipping Month Dice Menu + |

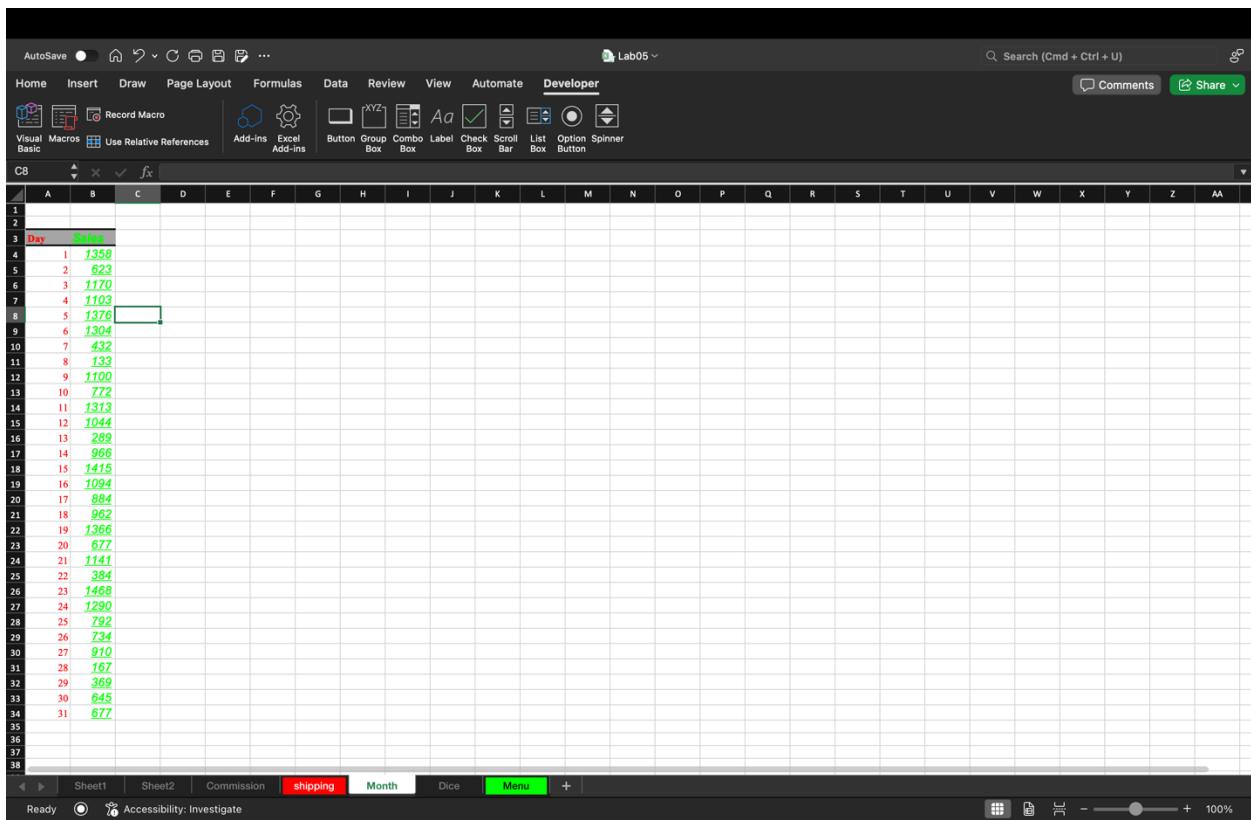
Ready Accessibility: Investigate 164%

B. Change cells in column B to Arial, size 14, green color, italic, and underline

Code:

```
Sub formatMonthColumnB()
'Lab 5 P7B
With Worksheets("Month").Range("B3:B34")
    .Font.Name = "Arial"
    .Font.Size = 14
    .Font.Color = RGB(0, 255, 0)
    .Font.Italic = True
    .Font.Underline = True
End With
End Sub
```

Result:



Day	Sales
1	1358
2	623
3	1170
4	1103
5	1376
6	1304
7	432
8	133
9	1100
10	772
11	1313
12	1044
13	289
14	966
15	1415
16	1094
17	884
18	962
19	1366
20	677
21	1141
22	384
23	1468
24	1290
25	792
26	734
27	910
28	167
29	369
30	645
31	677
32	
33	
34	

## 8. Using Dice sheet

A. Write VBA code that rolls a dice 100 times. For each roll, reward the player with a monetary value that is equal to the number shown on the dice

Code:

```
Sub rollDice()
'Lab 5 P8

Worksheets("Dice").Activate

For i = 2 To 101
    Roll = Int((6 * Rnd) + 1)
    Range("A" & i).Value = Roll
    Range("B" & i).Value = Roll
Next i
End Sub
```

Result:

Roll	Reward
1	5
2	4
3	4
4	4
5	2
6	2
7	5
8	1
9	5
10	5
11	5
12	1
13	3
14	6
15	5
16	3
17	6
18	6
19	1
20	6
21	3
22	4
23	5
24	1
25	4
26	3
27	2
28	4
29	4
30	2
31	2
32	5
33	5
34	4
35	6
36	6
37	2
38	5
39	6
40	2
41	4
42	1
43	6
44	5
45	1

Sub rollDice()  
'Lab 5 P8  
Worksheets("Dice").Activate  
For i = 2 To 101  
 Roll = Int((6 \* Rnd) + 1)  
 Range("A" & i).Value = Roll  
 Range("B" & i).Value = Roll  
Next i  
End Sub

## 9. Using Menu sheet

A. Write VBA code that takes customer order, fills the Excel sheet, adds cost of the order items, calculates the total cost of the order and displays it to the customer

Code:

```

Sub menuOrders()
'Lab 5 Problem 9
Worksheets("Menu").Select

'Declare Variables
Dim i As Integer
Dim orderStr As String
Dim orderSp() As String
Dim cost As Currency, total As Currency

'Name a line to go to in the case loop
start:
'Receive Customer order
orderStr = InputBox("Please Enter your Order: ")
MsgBox ("Your Order is: " & orderStr)

'Parse Order String
orderSp() = Split(orderStr, ",")  

total = 0

'Process order and calculate cost
For i = LBound(orderSp) To UBound(orderSp) Step 2
    Select Case orderSp(i)
        Case Is = "burger"
            cost = orderSp(i + 1) * Range("B2")
            Range("C2").Value = orderSp(i + 1)
            Range("D2").Value = cost
            total = total + cost
        Case Is = "fries"
            cost = orderSp(i + 1) * Range("B2")
            Range("C3").Value = orderSp(i + 1)
            Range("D3").Value = cost
            total = total + cost
        Case Is = "soda"

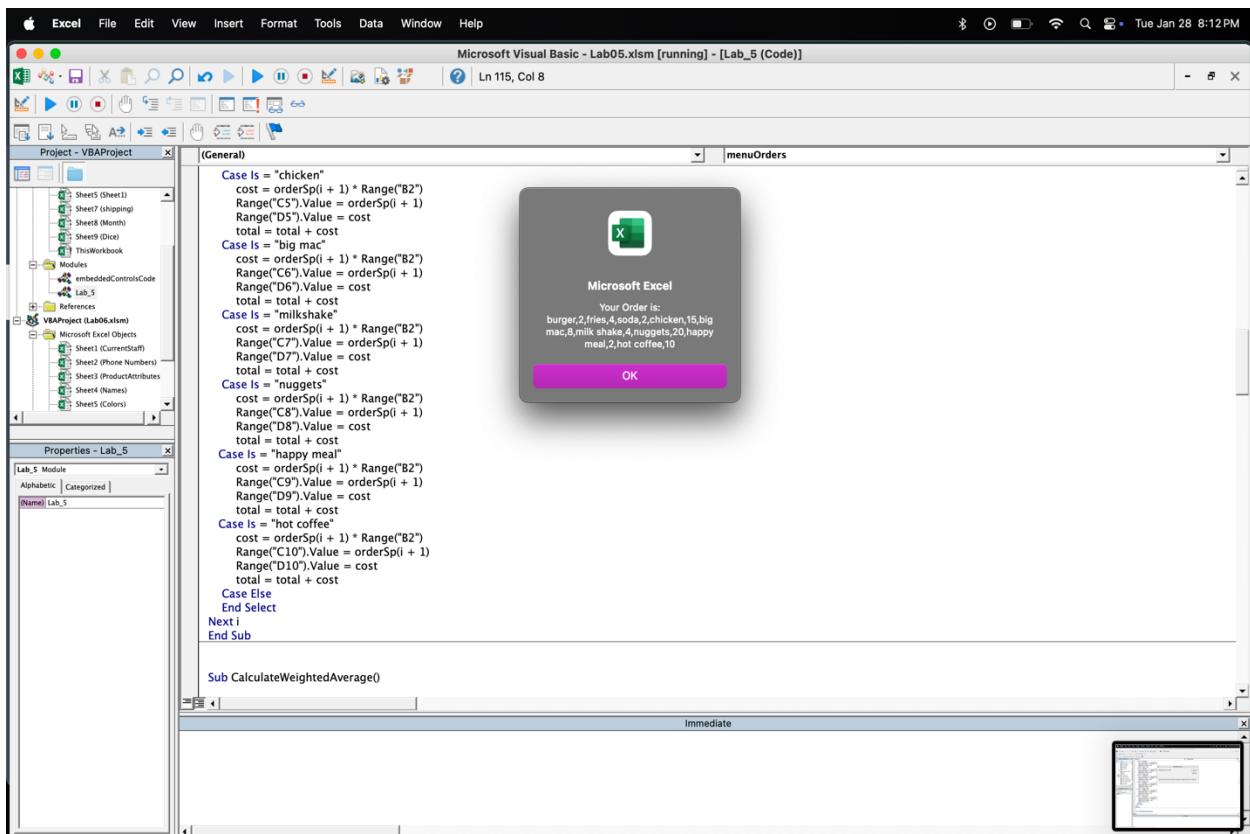
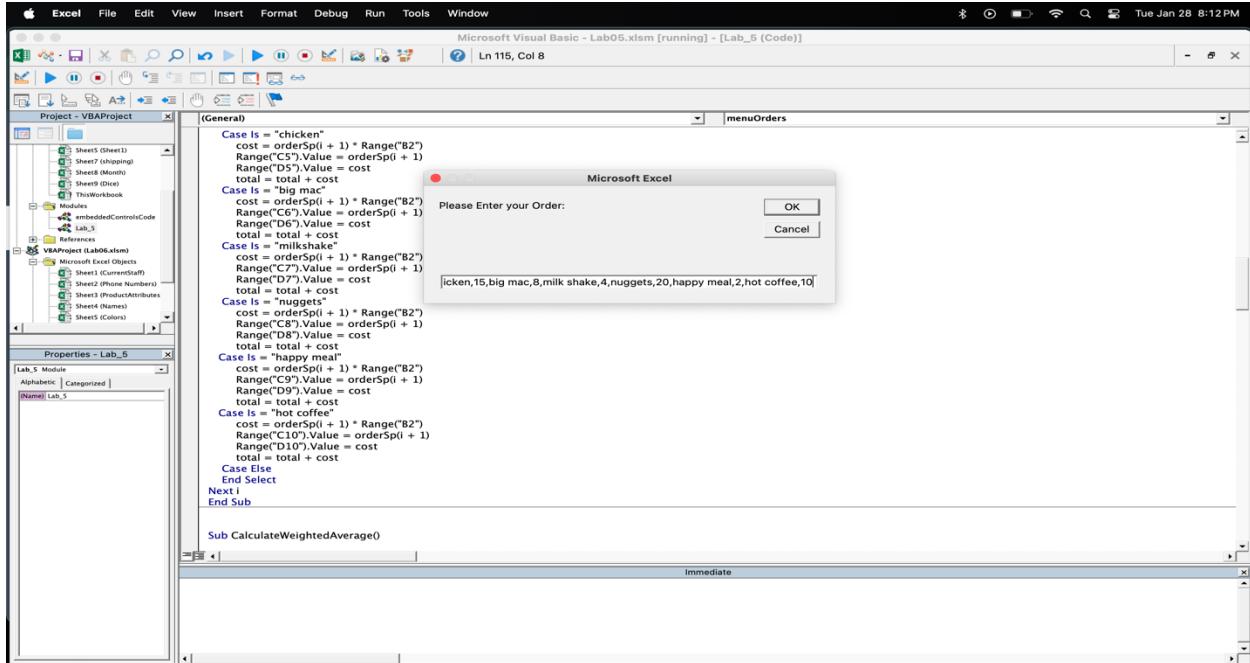
```

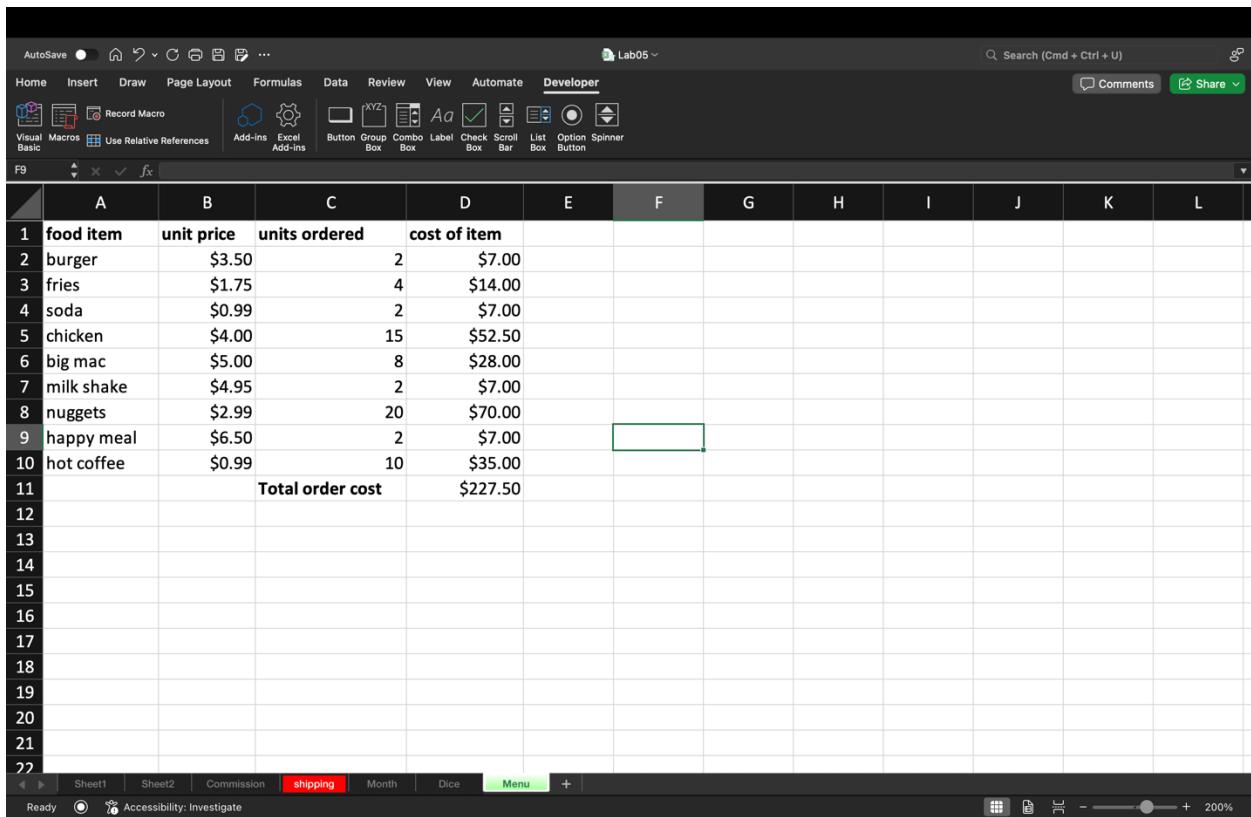
```

Case Is = "soda"
cost = orderSp(i + 1) * Range("B2")
Range("C4").Value = orderSp(i + 1)
Range("D4").Value = cost
total = total + cost
Case Is = "chicken"
cost = orderSp(i + 1) * Range("B2")
Range("C5").Value = orderSp(i + 1)
Range("D5").Value = cost
total = total + cost
Case Is = "big mac"
cost = orderSp(i + 1) * Range("B2")
Range("C6").Value = orderSp(i + 1)
Range("D6").Value = cost
total = total + cost
Case Is = "milkshake"
cost = orderSp(i + 1) * Range("B2")
Range("C7").Value = orderSp(i + 1)
Range("D7").Value = cost
total = total + cost
Case Is = "nuggets"
cost = orderSp(i + 1) * Range("B2")
Range("C8").Value = orderSp(i + 1)
Range("D8").Value = cost
total = total + cost
Case Is = "happy meal"
cost = orderSp(i + 1) * Range("B2")
Range("C9").Value = orderSp(i + 1)
Range("D9").Value = cost
total = total + cost
Case Is = "hot coffee"
cost = orderSp(i + 1) * Range("B2")
Range("C10").Value = orderSp(i + 1)
Range("D10").Value = cost
total = total + cost
Case Else
End Select

```

## Result:



**Output:**

The screenshot shows a Microsoft Excel spreadsheet titled "Lab05". The ribbon at the top has tabs for Home, Insert, Draw, Page Layout, Formulas, Data, Review, View, Automate, and Developer. The Developer tab is selected. The formula bar shows "F9". The main area contains a table with the following data:

	A	B	C	D	E	F	G	H	I	J	K	L
1	food item	unit price	units ordered	cost of item								
2	burger	\$3.50		2	\$7.00							
3	fries	\$1.75		4	\$14.00							
4	soda	\$0.99		2	\$7.00							
5	chicken	\$4.00		15	\$52.50							
6	big mac	\$5.00		8	\$28.00							
7	milk shake	\$4.95		2	\$7.00							
8	nuggets	\$2.99		20	\$70.00							
9	happy meal	\$6.50		2	\$7.00							
10	hot coffee	\$0.99		10	\$35.00							
11	<b>Total order cost</b>				<b>\$227.50</b>							
12												
13												
14												
15												
16												
17												
18												
19												
20												
21												
22												

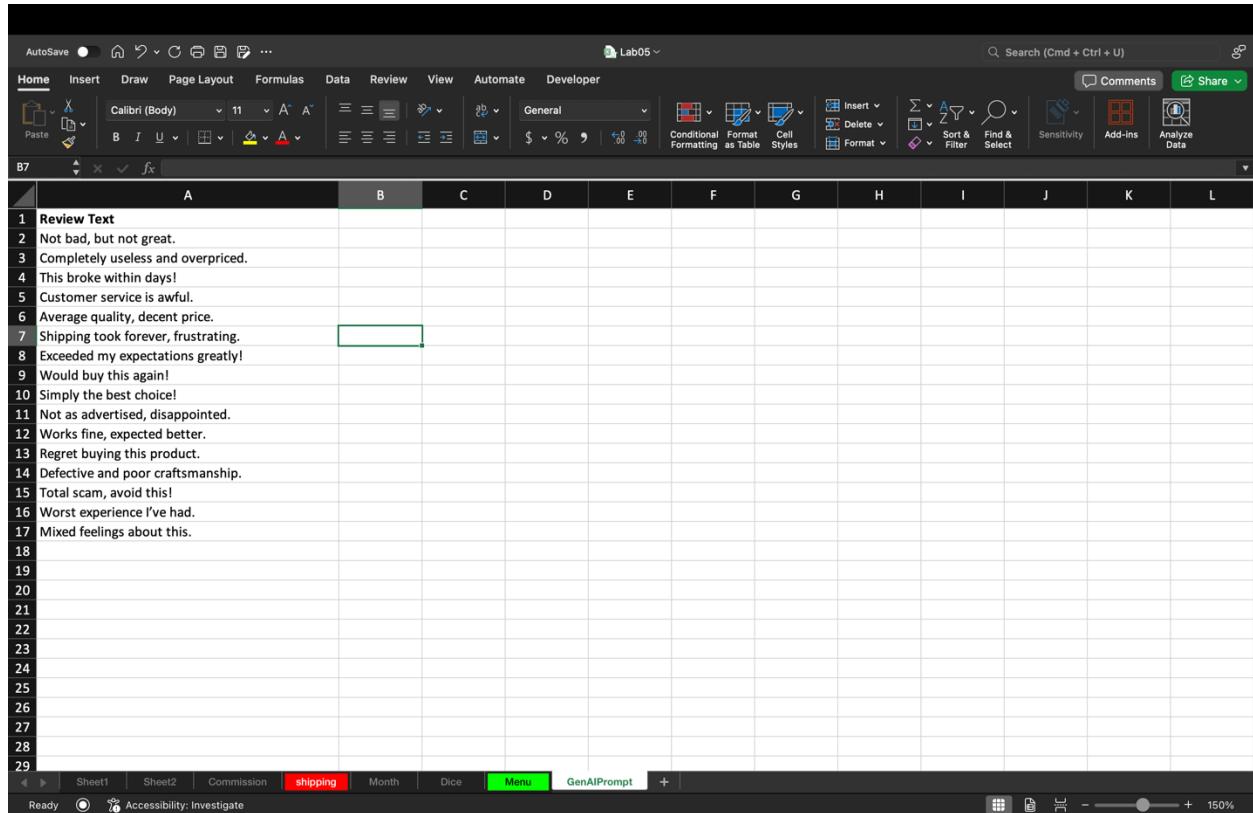
The ribbon shows tabs for Sheet1, Sheet2, Commission, shipping (which is highlighted in red), Month, Dice, and Menu. The status bar at the bottom indicates "Ready", "Accessibility: Investigate", and a zoom level of 200%.

## 10. GEN AI Prompt Engineering

### I am using Perplexity

#### A. Input Data

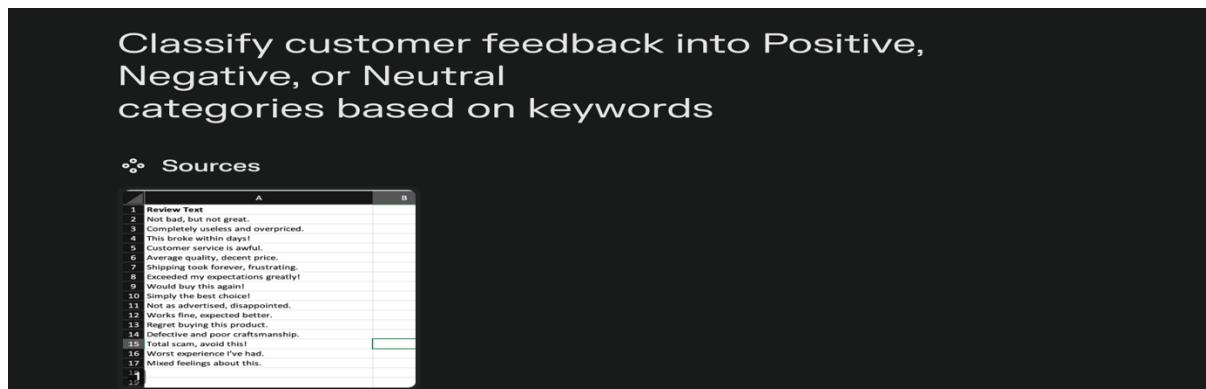
- Created a Sheet in Lab 05 named as “GenAIPrompt”
- Entered sample feedbacks



	A	B	C	D	E	F	G	H	I	J	K	L
1	Review Text											
2	Not bad, but not great.											
3	Completely useless and overpriced.											
4	This broke within days!											
5	Customer service is awful.											
6	Average quality, decent price.											
7	Shipping took forever, frustrating.											
8	Exceeded my expectations greatly!											
9	Would buy this again!											
10	Simply the best choice!											
11	Not advertised, disappointed.											
12	Works fine, expected better.											
13	Regret buying this product.											
14	Defective and poor craftsmanship.											
15	Total scam, avoid this!											
16	Worst experience I've had.											
17	Mixed feelings about this.											
18												
19												
20												
21												
22												
23												
24												
25												
26												
27												
28												
29												

#### B. AI Rule AI

Prompt:



Classify customer feedback into Positive, Negative, or Neutral categories based on keywords

Sources

1 Review Text
2 Not bad, but not great.
3 Completely useless and overpriced.
4 This broke within days!
5 Customer service is awful.
6 Average quality, decent price.
7 Shipping took forever, frustrating.
8 Exceeded my expectations greatly!
9 Would buy this again!
10 Simply the best choice!
11 Not as advertised, disappointed.
12 Works fine, expected better.
13 Regret buying this product.
14 Defective and poor craftsmanship.
15 Total scam, avoid this!
16 Worst experience I've had.
17 Mixed feelings about this.

GenAI:

The screenshot shows the Perplexity AI interface with the following content:

**Perplexity**

Let me analyze the customer reviews and categorize them based on sentiment:

**Positive Reviews**

- "Exceeded my expectations greatly!" (#8)
- "Would buy this again!" (#9)
- "Simply the best choice!" (#10)

**Negative Reviews**

- "Completely useless and overpriced." (#3)
- "This broke within days!" (#4)
- "Customer service is awful." (#5)
- "Shipping took forever, frustrating." (#7)
- "Defective and poor craftsmanship." (#14)
- "Total scam, avoid this!" (#15)
- "Worst experience I've had." (#16)
- "Regret buying this product." (#13)

**Neutral Reviews**

- "Not bad, but not great." (#2)
- "Average quality, decent price." (#6)
- "Not as advertised, disappointed." (#11)
- "Works fine, expected better." (#12)
- "Mixed feelings about this." (#17)

The sentiment analysis shows predominantly negative feedback, with 8 negative reviews,

- "Regret buying this product." (#13)

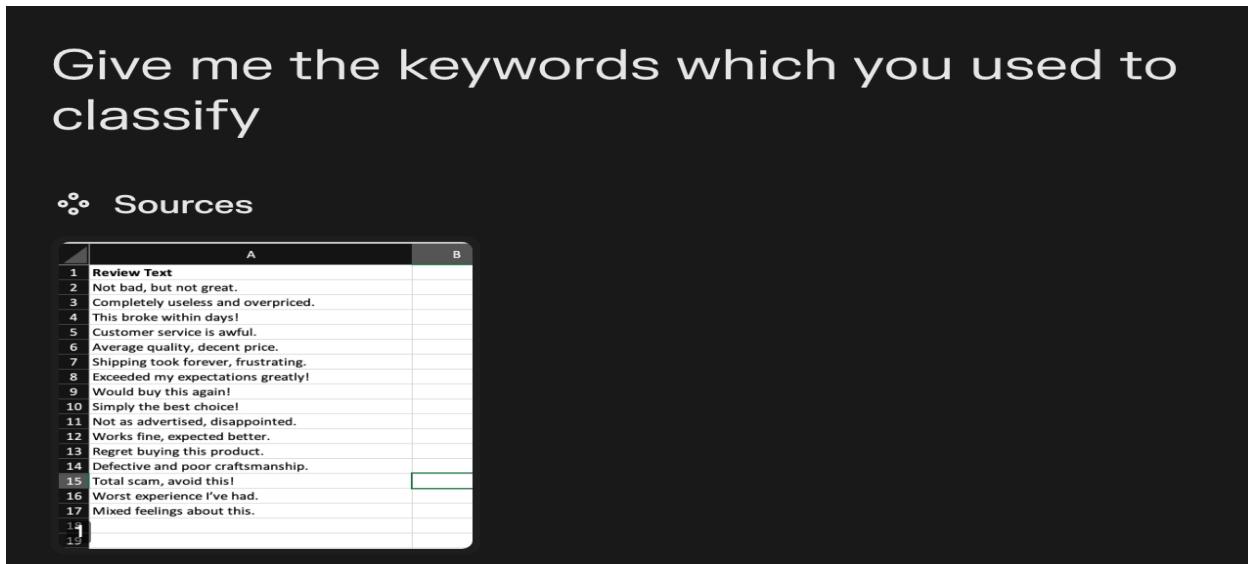
**Neutral Reviews**

- "Not bad, but not great." (#2)
- "Average quality, decent price." (#6)
- "Not as advertised, disappointed." (#11)
- "Works fine, expected better." (#12)
- "Mixed feelings about this." (#17)

The sentiment analysis shows predominantly negative feedback, with 8 negative reviews, 3 positive reviews, and 5 neutral reviews.

Share Rewrite

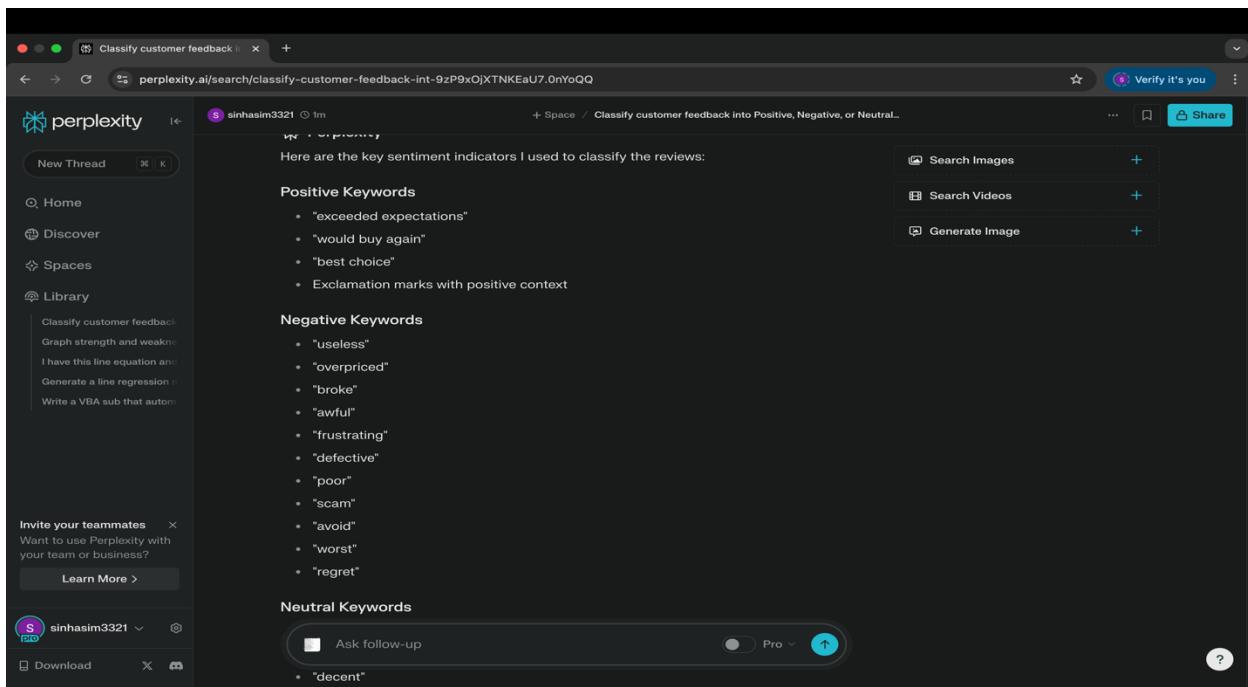
Prompt:



**Review Text**

- 1 Not bad, but not great.
- 2 Completely useless and overpriced.
- 3 This broke within days!
- 4 Customer service is awful.
- 5 Average quality, decent price.
- 6 Shipping took forever, frustrating.
- 7 Exceeded my expectations greatly!
- 8 Would buy this again!
- 9 Simply the best choice!
- 10 Not as advertised, disappointed.
- 11 Works fine, expected better.
- 12 Regret buying this product.
- 13 Defective and poor craftsmanship.
- 14 Total scam, avoid this!
- 15 Worst experience I've had.
- 16 Mixed feelings about this.
- 17

GenAI:



Here are the key sentiment indicators I used to classify the reviews:

**Positive Keywords**

- "exceeded expectations"
- "would buy again"
- "best choice"
- Exclamation marks with positive context

**Negative Keywords**

- "useless"
- "overpriced"
- "broke"
- "awful"
- "frustrating"
- "defective"
- "poor"
- "scam"
- "avoid"
- "worst"
- "regret"

**Neutral Keywords**

- "decent"

The screenshot shows a web browser window for perplexity.ai with the URL [perplexity.ai/search/classify-customer-feedback-int-9zP9xOjXTNKEaU7.OnYoQQ](https://perplexity.ai/search/classify-customer-feedback-int-9zP9xOjXTNKEaU7.OnYoQQ). The main content area displays two sections of keywords:

- Negative Keywords:**
  - "frustrating"
  - "defective"
  - "poor"
  - "scam"
  - "avoid"
  - "worst"
  - "regret"
- Neutral Keywords:**
  - "not bad, but not great"
  - "average"
  - "decent"
  - "fine"
  - "mixed feelings"
  - Phrases combining positive and negative elements
  - Words indicating moderate satisfaction/dissatisfaction

Below the keyword lists, a note states: "The classification also considers the overall context and tone of each review, not just individual keywords in isolation." At the bottom, there are "Share" and "Rewrite" buttons, along with a "Pro" toggle switch.

- Manually entering all the keywords and Category

### C. Dynamic Classification

Code:

```

-----  

Sub CategorizeSentiment()  

'Lab 5 P10
Dim ws As Worksheet
Dim lastRow As Long, i As Long
Dim text As String

Set ws = ActiveSheet
lastRow = ws.Range("A" & Rows.Count).End(xlUp).Row

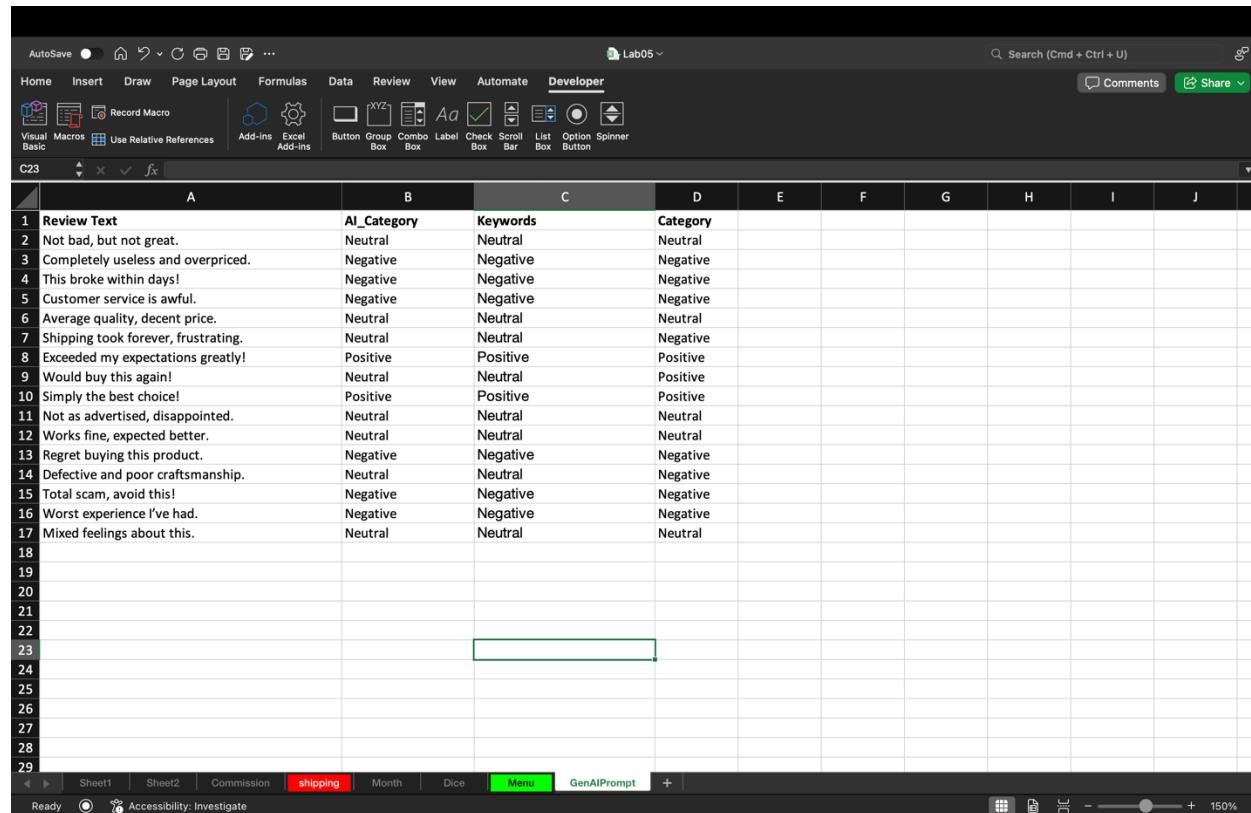
For i = 2 To lastRow
    text = LCase(ws.Range("A" & i).Value)

    'Positive keywords
    If InStr(text, "exceed") > 0 Or InStr(text, "best") > 0 Or _
        InStr(text, "buy again") > 0 Then
        ws.Range("B" & i).Value = "Positive"

    'Negative keywords
    ElseIf InStr(text, "useless") > 0 Or InStr(text, "broke") > 0 Or _
        InStr(text, "awful") > 0 Or InStr(text, "scam") > 0 Or _
        InStr(text, "worst") > 0 Or InStr(text, "regret") > 0 Then
        ws.Range("B" & i).Value = "Negative"

    'Neutral keywords
    Else
        ws.Range("B" & i).Value = "Neutral"
    End If
    Next i
End Sub|
```

Output:



The screenshot shows an Excel spreadsheet titled 'Lab05'. The table has four columns: A, B, C, and D. Column A contains review text, column B contains the 'AI\_Category' (either Neutral, Positive, or Negative), and column D contains the 'Category' (also either Neutral, Positive, or Negative). The data is as follows:

	A	B	C	D
1	Review Text	AI_Category	Keywords	Category
2	Not bad, but not great.	Neutral	Neutral	Neutral
3	Completely useless and overpriced.	Negative	Negative	Negative
4	This broke within days!	Negative	Negative	Negative
5	Customer service is awful.	Negative	Negative	Negative
6	Average quality, decent price.	Neutral	Neutral	Neutral
7	Shipping took forever, frustrating.	Neutral	Neutral	Negative
8	Exceeded my expectations greatly!	Positive	Positive	Positive
9	Would buy this again!	Neutral	Neutral	Positive
10	Simply the best choice!	Positive	Positive	Positive
11	Not as advertised, disappointed.	Neutral	Neutral	Neutral
12	Works fine, expected better.	Neutral	Neutral	Neutral
13	Regret buying this product.	Negative	Negative	Negative
14	Defective and poor craftsmanship.	Neutral	Neutral	Negative
15	Total scam, avoid this!	Negative	Negative	Negative
16	Worst experience I've had.	Negative	Negative	Negative
17	Mixed feelings about this.	Neutral	Neutral	Neutral
18				
19				
20				
21				
22				
23				
24				
25				
26				
27				
28				
29				

LAB 06

## 1. Using CurrentStaff sheet

A. Write VBA Code that converts Hire date to a short Date. Show result in Column J

Code:

```
Sub convert()
'Lab 6 P1
'select column F
Worksheets("CurrentStaff").Activate
Columns("F:F").Select

'copy to coulmn J

Columns("F:F").Copy Columns("J:J")
'convert Hire Date to short date in column J
Columns("J:J").Select
Selection.NumberFormat = "m/d/yyyy" 'convert long date to short date
Selection.NumberFormat = "dddd, mmmm dd, yyyy" 'convert short date to long date
```

End Sub

## Output:

## 2. Using PhoneNumbers sheet

### A. Create two macros that sort the phone numbers by Name or department

Original Data:

The screenshot shows an Excel spreadsheet with the following data:

	A	B	C	D	E	F	G	H	I	J	K
1		<b>Sort by Name</b>	<b>Sort by Department</b>								
2	Wood, Jim	Research/Development	(303) 798-5393								
3	Shortt, Karol	Research/Development	(312) 856-5111								
4	Purvis, John	Research/Development	(619) 632-9190								
5	Peck, Brian	Research/Development	(303) 572-5993								
6	McManus, Tim	Research/Development	(312) 856-5109								
7	Jackson, Nancy	Research/Development	(303) 581-3111								
8	Hartman, Elinor	Research/Development	(312) 856-5235								
9	Burr, Helen	Research/Development	(303) 273-7113								
10	Wagner, Melissa	Quality Control	(303) 273-7112								
11	Suppes, Beau	Quality Control	(303) 247-6805								
12	Snyder, Ron	Quality Control	(619) 794-4304								
13	Jackson, Matt	Quality Control	(312) 856-5110								
14	Zytniewski, Jon	Quality Assurance	(212) 692-6400								
15	Rezendes, Nancy	Quality Assurance	(800) 456-4679								
16	Powell, Jon	Quality Assurance	(619) 794-4303								
17	Woods, Gwen	Project & Contract Services	(619) 794-4302								
18	Larson, Karen	Project & Contract Services	(800) 456-4678								

Macros: Sortbyname() and Sortbydepartment()

```

Sub Sortbyname()
    '
    ' Sortbyname Macro
    '

    Range(Selection, Selection.End(xlDown)).Select
    ActiveWorkbook.Worksheets("Phone Numbers").Sort.SortFields.Clear
    ActiveWorkbook.Worksheets("Phone Numbers").Sort.SortFields.Add2 Key:=Range(_
        "A2:A74"), SortOn:=xlSortOnValues, Order:=xlAscending, DataOption:=_
        xlSortNormal
    With ActiveWorkbook.Worksheets("Phone Numbers").Sort
        .SetRange Range("A2:C74")
        .Header = xlGuess
        .MatchCase = False
        .Orientation = xlTopToBottom
        .SortMethod = xlPinYin
        .Apply
    End With
End Sub

```

```

Sub Sortbydepartment()
    ' Sortbydepartment Macro

    Range("B2").Select
    Range(Selection, Selection.End(xlDown)).Select
    ActiveWorkbook.Worksheets("Phone Numbers").Sort.SortFields.Clear
    ActiveWorkbook.Worksheets("Phone Numbers").Sort.SortFields.Add2 Key:=Range(_
        "B2:B74"), SortOn:=xlSortOnValues, Order:=xlAscending, DataOption:= _
        xlSortNormal
    With ActiveWorkbook.Worksheets("Phone Numbers").Sort
        .SetRange Range("A2:C74")
        .Header = xlGuess
        .MatchCase = False
        .Orientation = xlTopToBottom
        .SortMethod = xlPinYin
        .Apply
    End With
End Sub

```

Buttons to Sortbyname() and Sortbydepartment()

The screenshot shows a Microsoft Excel spreadsheet titled 'Lab06'. The ribbon is visible at the top with tabs like Home, Insert, Draw, etc. The main area contains a table of data with columns for Employee ID, Name, Department, and Phone Number. Row 1 contains two buttons: 'Sort by Name' and 'Sort by Department'. The 'Sort by Name' button is highlighted with a blue background.

	A	B	C	D	E	F	G	H	I	J	K
1	<b>Sort by Name</b>	<b>Sort by Department</b>									
2	Baker, Leslie	Admin Training	(312) 856-2843								
3	Basso, Cathy	Logistics	(303) 986-9900								
4	Benabe, Nancy	Peptide Chemistry	(619) 794-4298								
5	Boston, Roger	Process Development	(716) 224-7521								
6	Bradshaw, Jill	Logistics	(510) 842-5252								
7	Burr, Helen	Research/Development	(303) 273-7113								
8	Cahn, Chris	International Clinical Safety	(303) 830-5181								
9	Clark, Susan	ADC	(316) 356-6978								
10	Darling, Jim	Project & Contract Services	(303) 492-7264								
11	DePasse, Gregory	Compliance	(316) 356-6994								
12	Farnes, Terri	Peptide Chemistry	(303) 782-9648								
13	Fink, Karla	Manufacturing	(619) 794-4296								
14	Frazee, John	Engineering/Maintenance	(303) 443-9598								
15	Freeman, Bonnie	Compliance	(303) 830-5869								
16	Garcia, Nita	ADC	(303) 581-8320								
17	Gibbs, Peggy	Professional Training Group	(303) 492-6226								
18	Graczyk, David	Engineering/Maintenance	(316) 356-6995								
19	Gregory, Bill	Manufacturing	(303) 986-9901								
20	Hall, Eric	Pharmacokinetics	(619) 794-4299								

At the bottom, there are tabs for 'Current Staff', 'Phone Numbers', 'ProductAttributes', 'Profits', 'Names', 'Colors', 'Products', 'Capitalized', and a '+' button. The 'Profits' tab is currently selected. The status bar at the bottom right shows 'Ready', 'Accessibility: Investigate', and a zoom level of '185%'.

#### B. Use buttons to run them

Lab 06 - Advanced Excel Functions											
Employee Data Analysis											
Row ID	Name	Department	Phone Number	Capitalized Name	Length of Name	Count of Name	Sum of Phone	Average Phone	Max Phone	Min Phone	Product ID
	First Name	Last Name	Phone Number	Capitalized Name	Length	Count	Sum	Avg	Max	Min	Product ID
1	Sort by Name		Sort by Department								
2	Baker, Leslie	Admin Training	(312) 856-2843								
3	Basso, Cathy	Logistics	(303) 986-9900								
4	Benabe, Nancy	Peptide Chemistry	(619) 794-4298								
5	Boston, Roger	Process Development	(716) 224-7521								
6	Bradshaw, Jill	Logistics	(510) 842-5252								
7	Burr, Helen	Research/Development	(303) 273-7113								
8	Cahn, Chris	International Clinical Safety	(303) 830-5181								
9	Clark, Susan	ADC	(316) 356-6978								
10	Darling, Jim	Project & Contract Services	(303) 492-7264								
11	DePasse, Gregory	Compliance	(316) 356-6994								
12	Farnes, Terri	Peptide Chemistry	(303) 782-9648								
13	Fink, Karla	Manufacturing	(619) 794-4296								
14	Fraze, John	Engineering/Maintenance	(303) 443-9598								
15	Freeman, Bonnie	Compliance	(303) 830-5869								
16	Garcia, Nita	ADC	(303) 581-8320								
17	Gibbs, Peggy	Professional Training Group	(303) 492-6226								
18	Gracyk, David	Engineering/Maintenance	(316) 356-6995								
19	Gregory, Bill	Manufacturing	(303) 986-9901								
20	Hall, Eric	Pharmacokinetics	(619) 794-4299								
Current Staff											
Phone Numbers											
ProductAttributes											
Profits											
Names											
Colors											
Products											
Capitalized											
+											
Ready   CurrentStaff   Phone Numbers   ProductAttributes   Profits   Names   Colors   Products   Capitalized   +											
Count: 73											
Autosave											
File											
Home Insert Draw Page Layout Formulas Data Review View Automate Developer											
Visual Basic Macros Record Macro Use Relative References Add-ins Excel Add-ins Button Group Box Combo Box Label Check Box Scroll Bar List Box Option Spinner Button											
Lab 06 ~											
Search (Cmd + Ctrl + U)											
Comments Share											
A2											
B2											
Autosave											
File											
Home Insert Draw Page Layout Formulas Data Review View Automate Developer											
Visual Basic Macros Record Macro Use Relative References Add-ins Excel Add-ins Button Group Box Combo Box Label Check Box Scroll Bar List Box Option Spinner Button											
Lab 06 ~											
Search (Cmd + Ctrl + U)											
Comments Share											
A2											
B2											
Autosave											
File											
Home Insert Draw Page Layout Formulas Data Review View Automate Developer											
Visual Basic Macros Record Macro Use Relative References Add-ins Excel Add-ins Button Group Box Combo Box Label Check Box Scroll Bar List Box Option Spinner Button											
Lab 06 ~											
Search (Cmd + Ctrl + U)											
Comments Share											
A2											
B2											
Autosave											
File											
Home Insert Draw Page Layout Formulas Data Review View Automate Developer											
Visual Basic Macros Record Macro Use Relative References Add-ins Excel Add-ins Button Group Box Combo Box Label Check Box Scroll Bar List Box Option Spinner Button											
Lab 06 ~											
Search (Cmd + Ctrl + U)											
Comments Share											
A2											
B2											
Autosave											
File											
Home Insert Draw Page Layout Formulas Data Review View Automate Developer											
Visual Basic Macros Record Macro Use Relative References Add-ins Excel Add-ins Button Group Box Combo Box Label Check Box Scroll Bar List Box Option Spinner Button											
Lab 06 ~											
Search (Cmd + Ctrl + U)											
Comments Share											
A2											
B2											
Autosave											
File											
Home Insert Draw Page Layout Formulas Data Review View Automate Developer											
Visual Basic Macros Record Macro Use Relative References Add-ins Excel Add-ins Button Group Box Combo Box Label Check Box Scroll Bar List Box Option Spinner Button											
Lab 06 ~											
Search (Cmd + Ctrl + U)											
Comments Share											
A2											
B2											
Autosave											
File											
Home Insert Draw Page Layout Formulas Data Review View Automate Developer											
Visual Basic Macros Record Macro Use Relative References Add-ins Excel Add-ins Button Group Box Combo Box Label Check Box Scroll Bar List Box Option Spinner Button											
Lab 06 ~											
Search (Cmd + Ctrl + U)											
Comments Share											
A2											
B2											
Autosave											
File											
Home Insert Draw Page Layout Formulas Data Review View Automate Developer											
Visual Basic Macros Record Macro Use Relative References Add-ins Excel Add-ins Button Group Box Combo Box Label Check Box Scroll Bar List Box Option Spinner Button											
Lab 06 ~											
Search (Cmd + Ctrl + U)											
Comments Share											
A2											
B2											
Autosave											
File											
Home Insert Draw Page Layout Formulas Data Review View Automate Developer											
Visual Basic Macros Record Macro Use Relative References Add-ins Excel Add-ins Button Group Box Combo Box Label Check Box Scroll Bar List Box Option Spinner Button											
Lab 06 ~											

### 3. Using ProductAttributes sheet

A. Write VBA code that converts the layout shown on top of sheet (shown here also) to the one shown below

Code:

```

Sub organize()
'Lab 6 P3
'setup column labels (headers)
Worksheets("ProductAttributes").Activate
Range("D8").Value = Range("A3").Value
Range("E8").Value = "Country"
Range("F8").Value = "Value"
Range("D8").Font.Bold = True
Range("E8").Font.Bold = True
Range("F8").Font.Bold = True

'populate Product Attribute column
Range("D9").Value = Range("A4").Value
Range("D13").Value = Range("A5").Value
Range("D17").Value = Range("A6").Value

'populate Country column
j = 9
For i = 1 To 3
    Cells(j, 5).Value = "UK"
    Cells(j + 1, 5).Value = "FR"
    Cells(j + 2, 5).Value = "DE"
    j = j + 4
Next i

'populate Value column
For i = 1 To 3
    j = i + 8
    Cells(j, 6).Value = Cells(4, 2 + i).Value
    Cells(j + 4, 6).Value = Cells(5, 2 + i).Value
    Cells(j + 8, 6).Value = Cells(6, 2 + i).Value
Next i

End Sub

```

## Output:

AutoSave

Home Insert Draw Page Layout Formulas Data Review View Automate Developer

E25

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
1				Attribute value																				
2				England (UK)	France (FR)	Germany (DE)																		
3	product Attribute																							
4	Color	Red	Green	Yellow																				
5	Shape	Round	Rectangle	Diamond																				
6	Units	200	100	350																				
7																								
8		product Attribute	Country <sup>n</sup>	Value																				
9		Color	UK	Red																				
10			FR	Green																				
11			DE	Yellow																				
12																								
13		Shape	UK	Round																				
14			FR	Rectangle																				
15			DE	Diamond																				
16																								
17		Units	UK	200																				
18			FR	100																				
19			DE	350																				
20																								
21																								
22																								
23																								
24																								
25																								
26																								
27																								
28																								
29																								
30																								
31																								
32																								
33																								
34																								
35																								
36																								
37																								
38																								
39																								
40																								
41																								
42																								
43																								
44																								
45																								

CurrentStart Phone Numbers ProductAttributes Profits Names Colors Products Capitalized +

Accessibility: Investigate

Search (Cmd + Ctrl + U) Comments Share

Tue Jan 28 8:20 PM

## 4. Using Profits sheet

A. Write VBA code that calculates the totals in column H, averages in column I and the YTD Profits in row 6 and that highlights column H cells in yellow, Average cells in red and cells B6 – BG6 in cyan. Verify your results manually using Excel

Code:

```

Sub profit()
'Lab 6 P4
Dim total As Double

Worksheets("Profits").Activate

'calculate totals for column H cells
For i = 1 To 3
    total = 0
    For j = 1 To 6
        total = total + Cells(i + 2, j + 1)
    Next j
    Range("H" & (i + 2)).Value = total
    Range("H" & (i + 2)).Interior.color = RGB(255, 255, 0)
Next i

'calculate averages for column I
For i = 1 To 3
    Range("I" & (i + 2)).Value = Application.Average(Range("B" & i + 2, "G" & i + 2))
    Range("I" & (i + 2)).Interior.color = RGB(255, 0, 0)
Next i

'calculate YTD in row 6
total = 0

For i = 1 To 6
    total = total + Cells(5, i + 1).Value
    Cells(6, i + 1).Value = total
    Cells(6, i + 1).Interior.color = RGB(0, 255, 255)
Next i
End Sub

```

## Output:

## 5. Using Names sheet

### A. Write VBA code to organize the names list in a better way

Code:

```

Sub organizeNames()
    'Lab 6 Problem
    Dim arr() As Variant
    arr = Array("Name", "Company", "Office Phone")
    column = Array("D", "E", "F")
    'organize Names
    'set up columns header
    For i = 1 To 3
        Range(column(i) & "1").Value = arr(i - 1)
        Range(column(i) & "1").Font.Bold = True
        Range(column(i) & "1").Interior.Color = Range("A1").Interior.Color
    Next i

    'sort data
    j = 2
    totalRows = 1201
    For i = 1 To totalRows Step 3
        Range(column(0) & (i + 1)).Value = Range("A" & j).Value
        Range(column(1) & (i + 1)).Value = Range("A" & (j + 1)).Value
        Range(column(2) & (i + 1)).Value = Range("A" & (j + 2)).Value
        j = j + 3
    Next i

    'autofit columns sizes
    Columns("D:F").EntireColumn.AutoFit
End Sub

```

Output:

	A	B	C	D	E	F
1	Entry			Name	Company	Office Phone
2	Parker, Carlyn L			Parker, Carlyn L	RE/MAX Alliance	RE/MAX Alliance
3	RE/MAX Alliance					
4	(313) 850-1916 , (866) 820-6457					
5	Prince, Larry J			Prince, Larry J	RE/MAX Properties, Inc	RE/MAX Properties, Inc
6	RE/MAX Properties, Inc					
7	(719) 576-5000					
8	Arnold, Darlene Joyce			Arnold, Darlene Joyce	ERA Shields Real Estate	ERA Shields Real Estate
9	ERA Shields Real Estate					
10	(719) 593-1000 , (877) 888-8208					
11	Smith, Karen L			Smith, Karen L	Leeper & Associates	Leeper & Associates
12	Leeper & Associates					
13	(313) 469-6723 , (877) 626-1419					
14	Cunningham, Carrie A			Cunningham, Carrie A	RE/MAX Pueblo West Inc.	RE/MAX Pueblo West Inc.
15	RE/MAX Pueblo West Inc.					
16	(719) 547-1739 ext. 1739, (877) 699-1716					
17	Ward, Suzanne M			Ward, Suzanne M	Coldwell Banker	Coldwell Banker
18	Coldwell Banker					
19	(313) 320-5733 , (877) 856-8355					
20	Segreti, Margaret			Segreti, Margaret	A.H. Leffler & Assoc., LLC	A.H. Leffler & Assoc., LLC
21	A.H. Leffler & Assoc., LLC					
22	(970) 353-6088					
23	Barros, Christina de			Barros, Christina de	Agi Realty	Agi Realty
24	Agi Realty					
25	(970) 581-9667					
26	Ranes, Kim			Ranes, Kim	RE/MAX Horizons Group	RE/MAX Horizons Group
27	RE/MAX Horizons Group					
28	(813) 469-6723 , (877) 696-1419					

## 6. Using Colors sheet

A. Write VBA code to count the number of cells that have the same background color

Code:

```
Function countColor(rng As Range, ref_cell As Range) As Integer
'Lab 6 P6

Dim num As Integer
Dim rng_cell As Range
num = 0
For Each rng_cell In rng
    If rng_cell.Interior.ColorIndex = ref_cell.Interior.ColorIndex Then
        num = num + 1
    End If
Next
countColor = num
End Function
```

Output:

Name	Physics	Chemistry
Ron	37	87
Sam	88	48
Bob	79	85
Emma	76	44
Sara	82	72
Tim	95	69
8		
9 Color	Count	
10	3	
11	5	
12	4	
13		
14		
15		
16		
17		
18		
19		
20		
21		
22		
23		
24		
25		
26		
27		
28		
29		
30		
31		
32		
33		
34		
35		
36		
37		
38		
39		
40		
41		
42		
43		
44		
45		

## 7. Using Products sheet

A. Write VBA code to add the price of all Mobile and AC that were ordered

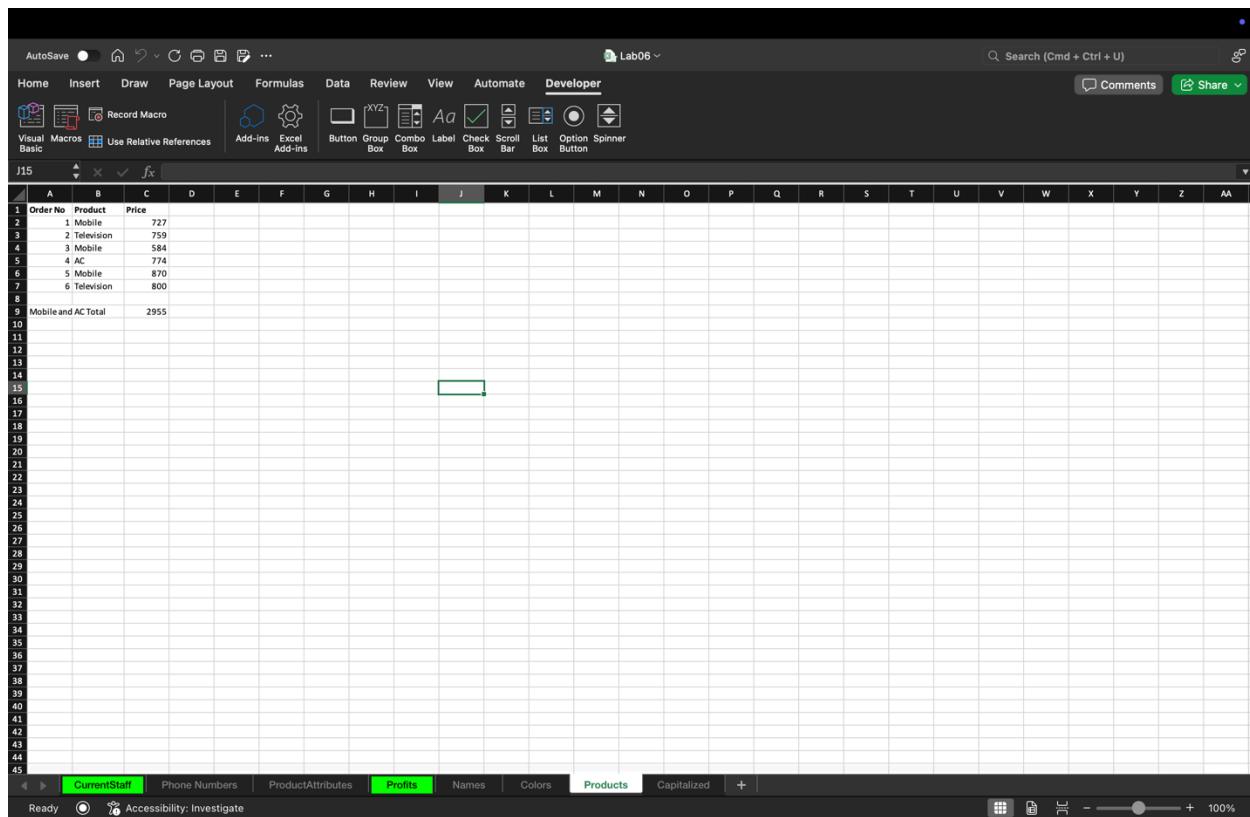
Code:

```
Sub totalPrice()
'Lab 6 P7

Dim total As Double
Dim rng_cell As Range

Set the_rng = Range("B2:B7")
total = 0
For Each rng_cell In the_rng
    If rng_cell.Value = "Mobile" Or rng_cell.Value = "AC" Then
        total = total + rng_cell.Offset(0, 1).Value
    End If
Next
Range("C9").Value = total
End Sub
```

Output:



The screenshot shows a Microsoft Excel spreadsheet titled 'Lab06'. The ribbon is visible at the top with tabs like Home, Insert, Draw, Page Layout, Formulas, Data, Review, View, Automate, and Developer. The Developer tab is selected. The formula bar shows the formula 'Range("C9").Value = total'. The main worksheet contains the following data:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	AA
1	Order No	Product	Price																								
2	1	Mobile	727																								
3	2	Television	759																								
4	3	Mobile	588																								
5	4	AC	774																								
6	5	Mobile	870																								
7	6	Television	800																								
8																											
9	Mobile and AC Total		2955																								
10																											
11																											
12																											
13																											
14																											
15																											
16																											
17																											
18																											
19																											
20																											
21																											
22																											
23																											
24																											
25																											
26																											
27																											
28																											
29																											
30																											
31																											
32																											
33																											
34																											
35																											
36																											
37																											
38																											
39																											
40																											
41																											
42																											
43																											
44																											
45																											

The ribbon shows the 'Products' tab is selected. The status bar indicates 'CurrentStaff' and 'Accessibility: Investigate'.

## 8. Using Capitalized sheet

A. Write VBA code to capitalize first letter of each name shown in column A. use input box to prompt users to select cell range. Display results in column B

Code:

```
Sub properNames0()
'Lab 6 P8
Dim rng As Range
Dim destRange As Range
Set rng = Application.InputBox("Select cell range", Type:=8)
Set destRange = Range("B" & rng.Row & ":B" & rng.Row + rng.Rows.Count - 1)
destRange.Value = Application.Proper(rng.Value)
End Sub
```

Output:

The screenshot shows an Excel spreadsheet titled "Lab06 – Saved to my Mac". The ribbon is set to the "Developer" tab. In the foreground, an "Input" dialog box is displayed, asking the user to "Select cell range". The background shows a table with two columns: "Name" and "Proper Name". The "Name" column lists names like Clark, Susan, Garcia, Nita, etc., while the "Proper Name" column shows them capitalized (Clark, Susan, Garcia, Nita, etc.). The "Developer" tab in the ribbon indicates that macros are enabled.

Name	Proper Name
1 clark, susan	Clark, Susan
2 garcia, nita	Garcia, Nita
4 johnson, gall	Johnson, Gall
5 baker, leslie	Baker, Leslie
6 huckstein, kathy	Huckstein, Kathy
7 martin, charles	Martin, Charles
8 poulet, tom	Poulet, Tom
9 kelly, steve	Kelly, Steve
10 kobus, jim	Kobus, Jim
11 degregory	Degregory
12 freeman, bonnie	Freeman, Bonnie
13 moore, susanne	Moore, Susanne
14 fraze, john	Fraze, John
15 graczyl, david	Graczyl, David
16 hamilton, neeta	Hamilton, Neeta
17 koch, annette	Koch, Annette
18 macleman, mike	Macleman, Mike
19 martinez, dave	Martinez, Dave
20 schiel, john	Schiel, John
21 mcmillen, michelle	McMillen, Michelle
22 williams, baron	Williams, Baron
23 kendall, joann	Kendall, Joann
24 montgomery, michelle	Montgomery, Michelle
25 perrotti, vince	Perrotti, Vince
26 cahn, chris	Cahn, Chris
27 moyers, kathy	Moyers, Kathy
28 shilling, kim	Shilling, Kim
29 basso, cathy	Basso, Cathy
30 bradshaw, jill	Bradshaw, Jill
31 hanson, sam	Hanson, Sam
32 krook, linda	Krook, Linda
33 pun, maria	Pun, Maria
34 turner, jerry	Turner, Jerry
35 wells, ann	Wells, Ann
36 fink, karla	Fink, Karla
37 gregory, bill	Gregory, Bill
38 hyner, sheryl	Hyner, Sheryl
39 lackner, lisa	Lackner, Lisa
40 stone, karen	Stone, Karen
41 riley, janice	Riley, Janice
42 tracy, pam	Tracy, Pam
43 willson, lena	Willson, Lena
44 benabie, nancy	Benabie, Nancy
45 farres, terri	Farres, Terri

Screenshot of Microsoft Excel showing a data manipulation process.

The top window (Sheet 1) displays a list of names from A1 to A73. A context menu is open over the range A1:A73, showing the "Select cell range" dialog with the address `$A$2:$A$73`.

The bottom window (Sheet 2) shows the same list of names from A1 to A73, with the first row (A1) containing the header "Name" and "Proper Name". The "Developer" tab is selected in both windows.

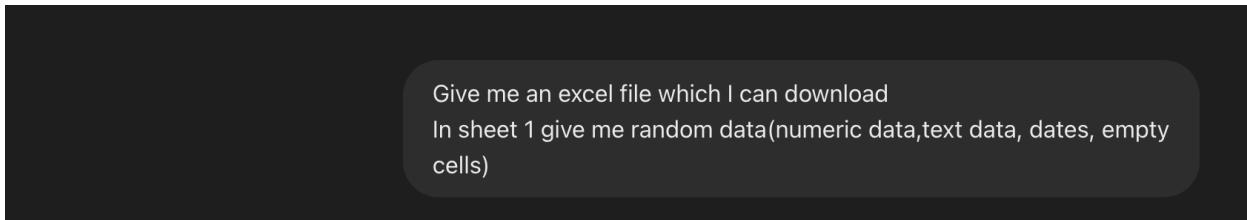
Both windows have tabs at the bottom: CurrentStaff, Phone Numbers, ProductAttributes, Profits, Names, Colors, Products, Capitalized, and a plus sign icon.

## **9. Gen AI and Prompt Engineering**

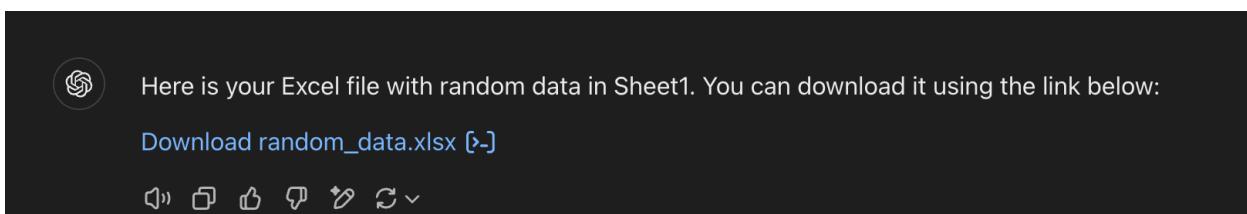
## I am using ChatGPT

## A. Scenario

### Prompt:



GenAI:



---

Excel Given By AI:

AutoSave

Home Insert Draw Page Layout Formulas Data Review View Automate Developer

GenAI\_Lab 06 – Saved to my Mac

Search (Cmd + Ctrl + U)

Comments Share

K15

A B C D E F G H I J K L M N C

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	C
1	Car Name	Company	Manufacture Date	Price (\$)	Horsepower										
2	Honda F-150	Chevrolet	2016-12-19 00:00:00	63871	305										
3	Mercedes SPortage	Kia	2004-07-06 00:00:00	27399	356										
4	Audi Malibu	Honda	2015-03-02 00:00:00	51214	363										
5	BMW Malibu	Chevrolet	2011-07-01 00:00:00	91416	560										
6	Nissan Camry	Kia	2023-02-14 00:00:00	75271	532										
7	Audi Altima	Hyundai	2007-01-01 00:00:00	49064	595										
8	Mercedes X5	BMW	2015-04-24 00:00:00	75091	271										
9	Ford A4	Toyota	2000-04-08 00:00:00	45818	128										
10	Audi Camry	Ford	2006-01-09 00:00:00	50525	591										
11	Hyundai A4	Toyota	2008-02-09 00:00:00	24830	96										
12	Honda X5	Mercedes	2013-08-13 00:00:00	22429	251										
13	BMW Civic	Mercedes	2012-02-20 00:00:00	11893	299										
14	Kia Civic	Chevrolet	2007-11-09 00:00:00	84909	237										
15	Honda X5	Hyundai	2002-08-14 00:00:00	52333	556										
16	Chevrolet X5	Hyundai	2022-11-13 00:00:00	8436	125										
17	Hyundai Camry	Honda	2015-08-09 00:00:00	79290	452										
18	Nissan Malibu	Honda	2019-01-05 00:00:00	81213	597										
19	Ford Altima	Honda	2012-04-20 00:00:00	10895	178										
20	Toyota Elantra	BMW	2013-06-12 00:00:00	24738	116										
21	Hyundai F-150	Nissan	2010-12-01 00:00:00	35746	359										
22															
23															
24															
25															
26															
27															
28															
29															
30															

Sheet1 +

Ready Accessibility: Good to go

144%

## B. Requirements and Output:

### 1. Dynamic Range Selection

2. Contextual Analysis: The macro should infer the data type and apply transformations based on the following rules:

#### - Numeric Data:

a. Manufacture Date: The original data is in YYYY-MM-DD and HH:MM:SS format. The AI has given me the Day and Days Since.

b. Price and HorsePower: The AI has given me Integer size and the square of the data.

#### - Text Data:

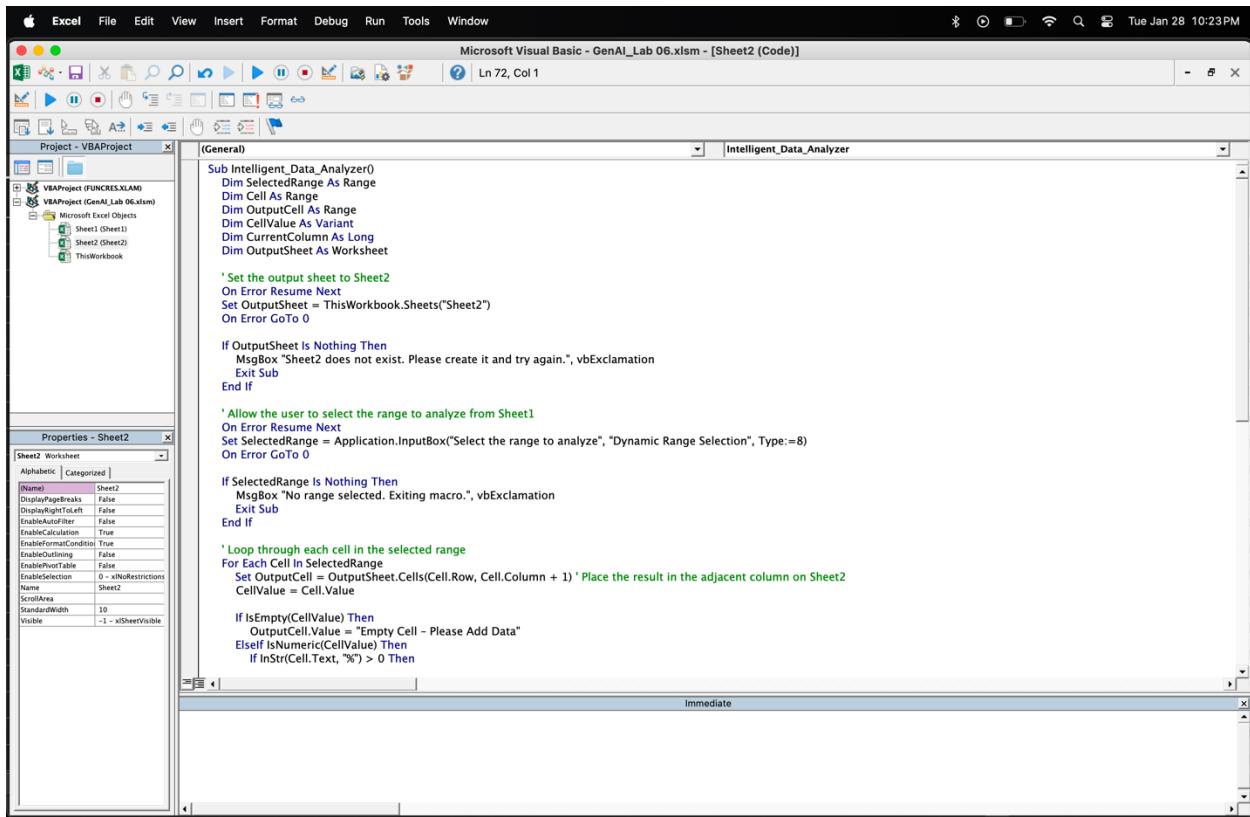
a. Car Name: The AI has given me the Uppercase and Word Count

b. Company: The AI has given me the Uppercase and Reversed Text

3. The Output is given adjacent to the original data in Sheet-2.

4. Error Handling: The Macro handles possible errors.

Code:



The screenshot shows the Microsoft Visual Basic Editor interface with the following details:

- Project Explorer:** Shows a VBAProject named "VBAProject (FuncRes.XLAM)" containing a module named "Intelligent\_Data\_Analyzer".
- Code Editor:** Displays the following VBA code:

```

Sub Intelligent_Data_Analyzer()
    Dim SelectedRange As Range
    Dim Cell As Range
    Dim OutputCell As Range
    DimCellValue As Variant
    Dim CurrentColumn As Long
    Dim OutputSheet As Worksheet

    ' Set the output sheet to Sheet2
    On Error Resume Next
    Set OutputSheet = ThisWorkbook.Sheets("Sheet2")
    On Error GoTo 0

    If OutputSheet Is Nothing Then
        MsgBox "Sheet2 does not exist. Please create it and try again.", vbExclamation
        Exit Sub
    End If

    ' Allow the user to select the range to analyze from Sheet1
    On Error Resume Next
    Set SelectedRange = Application.InputBox("Select the range to analyze", "Dynamic Range Selection", Type:=8)
    On Error GoTo 0

    If SelectedRange Is Nothing Then
        MsgBox "No range selected. Exiting macro.", vbExclamation
        Exit Sub
    End If

    ' Loop through each cell in the selected range
    For Each Cell In SelectedRange
        Set OutputCell = OutputSheet.Cells(Cell.Row, Cell.Column + 1) ' Place the result in the adjacent column on Sheet2
        CellValue = Cell.Value

        If IsEmpty(CellValue) Then
            OutputCell.Value = "Empty Cell - Please Add Data"
        ElseIf IsNumeric(CellValue) Then
            If InStr(Cell.Text, "%") > 0 Then
                OutputCell.Value = CellValue * CellValue
            Else
                OutputCell.Value = CellValue
            End If
        End If
    Next Cell
End Sub

```

- Properties Window:** Shows the properties for "Sheet2" with the following settings:
 

Name	Sheet2
DisplayPageBreaks	False
DisplayRightToLeft	False
EnableAutoFilter	False
EnableFormatConditions	True
EnableOutline	False
EnablePivotTable	False
HasPrintArea	0 ->NoRestrictions
Name	Sheet2
ScrollArea	
StandardWidth	10
Visible	-1 - xlSheetVisible
- Immediate Window:** Shows the text "Immediate" at the bottom.

The screenshot shows the Microsoft Visual Basic Editor interface. The menu bar includes Apple, Excel, File, Edit, View, Insert, Format, Debug, Run, Tools, and Window. The title bar reads "Microsoft Visual Basic - GenAI\_Lab 06.xlsxm - [Sheet2 (Code)]". The status bar at the bottom right shows "Tue Jan 28 10:24 PM".

The left pane displays the Project Explorer with "VBAProject (FUNCTIONS.XLAM)" and "VBAProject (GenAI\_Lab 06.xlsxm)" selected. Under "VBAProject (GenAI\_Lab 06.xlsxm)", "Sheet2 (Code)" is listed. The Properties window on the left shows settings for "Sheet2": Name = Sheet2, Category = Worksheet, and StandardWidth = 10.

The main code editor contains the following VBA code:

```
Set OutputCell = OutputSheet.Cells(Cell.Row, Cell.Column + 1) ' Place the result in the adjacent column on Sheet2
CellValue = Cell.Value

If IsEmpty(CellValue) Then
    OutputCell.Value = "Empty Cell - Please Add Data"
ElseIf IsNumeric(CellValue) Then
    If InStr(Cell.Text, "%") > 0 Then
        OutputCell.Value = "Percentage Value"
    ElseIf CellValue = Int(CellValue) Then
        'Integer check
        If CellValue < 10 Then
            OutputCell.Value = "Small Integer"
        ElseIf CellValue < 100 Then
            OutputCell.Value = "Medium Integer"
        Else
            OutputCell.Value = "Large Integer"
        End If
    Else
        OutputCell.Value = OutputCell.Value & ", Square: " & CellValue ^ 2
    End If
Else
    'Decimal check
    OutputCell.Value = "Decimal; Square Root: " & Sqr(CellValue)
End If

ElseIf IsDate(CellValue) Then
    ' Date analysis
    OutputCell.Value = "Day: " & Format(CellValue, "ddddd") & "; Days Since: " & Date - CellValue
ElseIf VarType(CellValue) = vbString Then
    'Text analysis
    If InStr(CellValue, ".") = 0 Then
        OutputCell.Value = "Reversed Text: " & StrReverse(CellValue)
    Else
        OutputCell.Value = "Word Count: " & UBound(Split(CellValue, " ")) + 1
    End If
    OutputCell.Value = OutputCell.Value & "; Uppercase: " & UCase(CellValue)
End If

'Handle unexpected data types
OutputCell.Value = "Unexpected Data Type"
End If
```

## Output: