Sub lotto\_winner()

' Create variables to hold winners. (Use "Long" because the numbers exceed the limit for integers)

Dim first\_place As Long

Dim second\_place As Long

Dim third\_place As Long

Dim runner1 As Long

Dim runner2 As Long

Dim runner3 As Long

' Establish the winning ticket numbers

first\_place = 3957481

second\_place = 5865187

third\_place = 2817729

' Establish the runner-up numbers

runner1 = 2275339

runner2 = 5868182

runner3 = 1841402

' Loop through each of the lotto tickets

For i = 1 To 1001

' Check if the lotto number matches the first place winner...

If Cells(i, 3).Value = first\_place Then

' If so, create a message box specifying the first place win

MsgBox " Congratulations " + Cells(i, 1).Value

' Retrieve the values associated with the winner and enter them into the winner's box.

Cells(2, 6).Value = Cells(i, 1).Value

Cells(2, 7).Value = Cells(i, 2).Value

Cells(2, 8).Value = first\_place

' Check if the lotto number matches the second place winner...

ElseIf Cells(i, 3).Value = second\_place Then

' Retrieve the values associated with the winner and enter them into the winner's box.

Cells(3, 6).Value = Cells(i, 1).Value

Cells(3, 7).Value = Cells(i, 2).Value

Cells(3, 8).Value = second\_place

' Check if the lotto number matches the second place winner...

ElseIf Cells(i, 3).Value = third\_place Then

' Retrieve the values associated with the winner and enter them into the winner's box.

Cells(4, 6).Value = Cells(i, 1).Value

Cells(4, 7).Value = Cells(i, 2).Value

Cells(4, 8).Value = third\_place

' Ends this series of IF/ELSE conditionals

End If

Next i

' Loop through the lotto tickets a second time to find the first instance of a "runner-up" winner

For i = 1 To 1001

' BONUS: Check for runner ups with an OR operator

If Cells(i, 3).Value = runner1 Or Cells(i, 3).Value = runner2 Or Cells(i, 3).Value = runner3 Then

' Retrieve the values associated with the winner and enter them into the winner's box.

runner\_up = Cells(i, 3).Value

Cells(5, 6).Value = Cells(i, 1).Value

Cells(5, 7).Value = Cells(i, 2).Value

Cells(5, 8).Value = runner\_up

' If first match is found, exit the for loop

Exit For

End If

Next i

End Sub

**06-Stu\_CreditCardChecker-CellComparison**

Sub creditCard():

Dim i As Integer ' row number

Dim cell\_val As Integer ' contents of column C

Dim cc\_total As Long ' what is going to go in column H

Dim credit\_card As String ' what is going to go in column G

Dim k As Integer ' leaderboard row

cc\_total = 0

k = 2

For i = 2 To 101:

cell\_val = Cells(i, 3).Value

credit\_card = Cells(i, 1).Value

' LOOP rows 2 to 101

' check if next row credit card is DIFFERENT

' if the same, then we only need to add to the cc\_total

' if DIFFERENT, then we need add last row, write out to the leaderboard

' reset the cc\_total to 0

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

' OMG we have a different credit card, panic

cc\_total = cc\_total + cell\_val

Cells(k, 7).Value = credit\_card

Cells(k, 8).Value = cc\_total

' reset

cc\_total = 0

k = k + 1

Else

' we just add to the total

cc\_total = cc\_total + cell\_val

End If

Next i

End Sub

**07-Stu\_Census\_Pt1**

' Steps:

' ----------------------------------------------------------------------------

' Part I:

' 1. Extract the number before the phrase "\_census\_data" to figure out the year.

' 2. Add the year to the first column of each spreadsheet.

' 3. Split the "Place" column into "County" and "State".

' 4. Convert the household and per capita income columns to currency values for all cells.

Sub Census\_pt1()

' --------------------------------------------

' LOOP THROUGH ALL SHEETS

' --------------------------------------------

For Each ws In Worksheets

' --------------------------------------------

' INSERT THE YEAR

' --------------------------------------------

' Create a Variable to Hold File Name, Last Row, and Year

Dim WorksheetName As String

' Determine the Last Row

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

' Grabbed the WorksheetName

WorksheetName = ws.Name

'MsgBox WorksheetName

' Split the WorksheetName

CensusYear = Split(WorksheetName, "\_")

'MsgBox CensusYear(0)

' Add a Column for the Year

ws.Range("A1").EntireColumn.Insert

' Add the word Year to the First Column Header

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

' Add the Year to all rows

ws.Range("A2:A" & LastRow) = CensusYear(0)

' --------------------------------------------

' SPLIT COUNTY AND STATE

' --------------------------------------------

' Add the State Column after County

ws.Range("C1").EntireColumn.Insert

' Rename Place to County

ws.Cells(1, 2).Value = "County"

' Label State Column

ws.Cells(1, 3).Value = "State"

' Split County and State and store values in appropriate

' column by looping through and renaming each

For i = 2 To LastRow

CountyState = ws.Cells(i, 2).Value

CSSplit = Split(CountyState, ", ")

'MsgBox CSSplit(1)

ws.Cells(i, 2).Value = CSSplit(0)

ws.Cells(i, 3).Value = CSSplit(1)

' MsgBox Cells(i, 2)

' MsgBox CSSplit(0)

Next i

' --------------------------------------------

' CORRECT THE CURRENCY FORMAT

' --------------------------------------------

' Add the currency

For i = 2 To LastRow

' For columns Household and Per Capita Income only

For j = 6 To 7

ws.Cells(i, j).Style = "Currency"

Next j

Next i

' --------------------------------------------

' FIXES COMPLETE

' --------------------------------------------

Next ws

MsgBox ("Fixes Complete")

End Sub

**08-Stu\_Census\_Pt2**

' Steps:

' ----------------------------------------------------------------------------

' Part II:

' 1. Loop through every worksheet and select the year contents.

' 2. Copy the year contents and paste it into the Combined\_Data tab

Sub Census\_pt2()

' Steps:

' ----------------------------------------------------------------------------

' Part II:

' 1. Loop through every worksheet and select the year contents.

' 2. Copy the year contents and paste it into the Combined\_Data tab

Sub Census\_pt2()

' Add a sheet named "Combined Data"

Sheets.Add.Name = "Combined\_Data"

'move created sheet to be first sheet

Sheets("Combined\_Data").Move Before:=Sheets(1)

' Specify the location of the combined sheet

Set combined\_sheet = Worksheets("Combined\_Data")

' Loop through all sheets

For Each ws In Worksheets

' Find the last row of the combined sheet after each paste

' Add 1 to get first empty row

lastRow = combined\_sheet.Cells(Rows.Count, "A").End(xlUp).Row + 1

' Find the last row of each worksheet

' Subtract one to return the number of rows without header

lastRowYear = ws.Cells(Rows.Count, "A").End(xlUp).Row - 1

' Copy the contents of each year sheet into the combined sheet

combined\_sheet.Range("A" & lastRow & ":K" & ((lastRowYear - 1) + lastRow)).Value = ws.Range("A2:K" & (lastRowYear + 1)).Value

Next ws

' Copy the headers from sheet 1

combined\_sheet.Range("A1:K1").Value = Sheets(2).Range("A1:K1").Value

' Autofit to display data

combined\_sheet.Columns("A:K").AutoFit

End Sub

From xpert learning

Sub ApplyConditionalFormatting()

Dim ws As Worksheet

Dim rng As Range

Dim cell As Range

' Set the worksheet where you want to apply conditional formatting

Set ws = ThisWorkbook.Sheets("Sheet1")

' Set the range of cells where you want to apply conditional formatting

Set rng = ws.Range("A1:A10")

' Loop through each cell in the range and apply conditional formatting based on the cell value

For Each cell In rng

If cell.Value > 5 Then

cell.Interior.Color = RGB(255, 0, 0) ' Set cell color to red if value is greater than 5

Else

cell.Interior.Color = RGB(0, 255, 0) ' Set cell color to green if value is 5 or less

End If

Next cell

End Sub

Sub ApplyConditionalFormatting()

Dim ws As Worksheet

Dim rng As Range

Dim condition1 As FormatCondition

Dim condition2 As FormatCondition

' Set the worksheet where you want to apply conditional formatting

Set ws = ThisWorkbook.Sheets("Sheet1")

' Set the range of cells where you want to apply conditional formatting

Set rng = ws.Range("A1:A10")

' Add the first conditional formatting rule (e.g., highlight cells greater than 5 with red color)

Set condition1 = rng.FormatConditions.Add(Type:=xlCellValue, Operator:=xlGreater, Formula1:="5")

With condition1

.Interior.Color = RGB(255, 0, 0) ' Set cell color to red

End With

' Add the second conditional formatting rule (e.g., highlight cells less than or equal to 5 with green color)

Set condition2 = rng.FormatConditions.Add(Type:=xlCellValue, Operator:=xlLessEqual, Formula1:="5")

With condition2

.Interior.Color = RGB(0, 255, 0) ' Set cell color to green

End With

End Sub