# VBA Coding Resouce for CTIC Spreadsheets

Sub UpdateInventorySheets()

' Turn off screen updating for performance

Application.ScreenUpdating = False1

' Error handling setup

On Error GoTo ErrorHandler

' Define worksheets, ranges, and other variables

Dim wsSource As Worksheet, wsOwlRef As Worksheet, wsTVSN As Worksheet

Dim lastRow As Long

Dim i As Integer

' Set references to the worksheets

Set wsSource = ThisWorkbook.Sheets("TVs, Owls, PTZs (2024)")

Set wsOwlRef = ThisWorkbook.Sheets("Meeting Owl Reference Sheet")

Set wsTVSN = ThisWorkbook.Sheets("TV Serial Numbers")

' Create Sheet 3 if it does not exist

Call CreateSheetIfMissing("TV Serial Numbers")

' Clear existing data in wsOwlRef and wsTVSN

wsOwlRef.Cells.ClearContents

wsTVSN.Cells.ClearContents

' Copy data from wsSource to wsOwlRef and wsTVSN with no gaps

Call CopyDataWithNoGaps(wsSource, wsOwlRef, Array("E", "K", "P"))

Call CopyDataWithNoGaps(wsSource, wsTVSN, Array("E", "F", "G", "H", "J"))

' Format the tables in wsOwlRef and wsTVSN

Call FormatAsTable(wsOwlRef, "TableStyleDark9")

Call FormatAsTable(wsTVSN, "TableStyleMedium2")

' Turn on screen updating

Application.ScreenUpdating = True

' Inform the user of completion

MsgBox "Update complete.", vbInformation

Exit Sub

ErrorHandler:

' Inform the user there was an error

MsgBox "An error occurred: " & Err.Description, vbCritical

' Turn on screen updating

Application.ScreenUpdating = True

End Sub

' Helper function to create a sheet if it doesn't exist

Sub CreateSheetIfMissing(sheetName As String)

Dim sheetExists As Boolean

sheetExists = False

On Error Resume Next

sheetExists = Not ThisWorkbook.Sheets(sheetName) Is Nothing

On Error GoTo 0

If Not sheetExists Then

ThisWorkbook.Sheets.Add(After:=ThisWorkbook.Sheets(ThisWorkbook.Sheets.Count)).Name = sheetName

End If

End Sub

' Helper function to copy data without gaps

Sub CopyDataWithNoGaps(wsSource As Worksheet, wsDestination As Worksheet, colAddresses As Variant)

Dim r As Range

Dim cIndex As Integer

Dim lastRow As Long

Dim destRow As Long

destRow = 1

For cIndex = LBound(colAddresses) To UBound(colAddresses)

lastRow = wsSource.Cells(wsSource.Rows.Count, colAddresses(cIndex)).End(xlUp).Row

For Each r In wsSource.Range(wsSource.Cells(2, colAddresses(cIndex)), wsSource.Cells(lastRow, colAddresses(cIndex)))

If Application.CountA(r.EntireRow) <> 0 Then

r.Copy

wsDestination.Cells(destRow, cIndex + 1).PasteSpecial Paste:=xlPasteValues

destRow = destRow + 1

End If

Next r

Next cIndex

Application.CutCopyMode = False

End Sub

' Helper function to format a worksheet as a table

Sub FormatAsTable(ws As Worksheet, tableStyle As String)

Dim tblRange As Range

Set tblRange = ws.Range("A1").CurrentRegion

ws.ListObjects.Add(xlSrcRange, tblRange, , xlYes).TableStyle = tableStyle

End Sub