Sub ExtractAndOrganizeData()

Dim wbSource As Workbook

Dim wsSource As Worksheet

Dim wsNew As Worksheet

Dim lastRow As Long

Dim companyName As String

Dim dateFilter As String

Dim yearFolder As String

Dim monthFile As String

Dim savePath As String

Dim mainFolder As String

Dim dict As Object

Dim key As Variant

Dim i As Long

Dim cell As Range

Dim rng As Range

Dim filteredData As Variant

Dim outputRow As Long

Dim year As String

Dim month As String

Dim monthName As String

On Error GoTo ErrorHandler ' Enable error handling

' Open the Schedule.xlsx file from the Desktop

Set wbSource = Workbooks.Open(Environ("USERPROFILE") & "\Desktop\Schedule.xlsx")

Set wsSource = wbSource.Sheets("Sheet1") ' Replace "Sheet1" with the correct worksheet name

companyName = "TOTAL OIL GH LIMITED\* DUNKWA)"

' Create a dictionary to store unique year-month combinations

Set dict = CreateObject("Scripting.Dictionary")

' Find the last row in the source worksheet

lastRow = wsSource.Cells(wsSource.Rows.Count, "A").End(xlUp).Row

MsgBox "Last row in the data: " & lastRow ' Debugging: Show last row

' Check if there is any data (excluding header row)

If lastRow <= 1 Then

MsgBox "No data found in the worksheet!", vbExclamation

Exit Sub

End If

' Loop through the data to filter and organize (start from Row 2)

For Each cell In wsSource.Range("A2:A" & lastRow)

' Use Trim and UCase to standardize comparison

If Trim(UCase(cell.Value)) = UCase(companyName) Then

dateFilter = wsSource.Cells(cell.Row, "B").Value ' Get the YYYYMM value

If Not dict.exists(dateFilter) Then

dict.Add dateFilter, cell.Row

End If

End If

Next cell

' Check if any data was found

If dict.Count = 0 Then

MsgBox "No data found for the company: " & companyName, vbExclamation

Exit Sub

End If

' Create the main folder inside the "New" folder on Desktop

mainFolder = Environ("USERPROFILE") & "\Desktop\New\TOTAL OIL GH LIMITED"

' Check if "New" folder exists, create if it doesn't

If Dir(Environ("USERPROFILE") & "\Desktop\New", vbDirectory) = "" Then

MkDir Environ("USERPROFILE") & "\Desktop\New"

End If

' Create the STALLION folder

If Dir(mainFolder, vbDirectory) = "" Then

MkDir mainFolder

End If

' Loop through each year-month in the dictionary

For Each key In dict.keys

year = Left(key, 4) ' Extract year (first 4 characters)

month = Right(key, 2) ' Extract month (last 2 characters)

' Convert month number to month name (e.g., "01" -> "jan")

monthName = LCase(Format(DateSerial(year, month, 1), "mmm"))

' Create year folder

yearFolder = mainFolder & "\" & year

If Dir(yearFolder, vbDirectory) = "" Then

MkDir yearFolder

End If

' Create a new workbook for the month

Set wsNew = Workbooks.Add.Sheets(1)

wsNew.Name = "Data"

' Add headers

wsNew.Cells(1, 1).Value = "ssnit"

wsNew.Cells(1, 2).Value = "name"

wsNew.Cells(1, 3).Value = "salary"

' Filter and copy data

outputRow = 2

For Each cell In wsSource.Range("A2:A" & lastRow)

If Trim(UCase(cell.Value)) = UCase(companyName) And wsSource.Cells(cell.Row, "B").Value = key Then

wsNew.Cells(outputRow, 1).Value = wsSource.Cells(cell.Row, "I").Value ' SSNIT

wsNew.Cells(outputRow, 2).Value = wsSource.Cells(cell.Row, "F").Value ' Name

wsNew.Cells(outputRow, 3).Value = wsSource.Cells(cell.Row, "L").Value ' Salary

outputRow = outputRow + 1

End If

Next cell

' Save the new workbook

monthFile = yearFolder & "\" & monthName & year & ".xlsx"

wsNew.Parent.SaveAs Filename:=monthFile, FileFormat:=xlOpenXMLWorkbook

wsNew.Parent.Close SaveChanges:=False

Next key

' Close the source workbook

wbSource.Close SaveChanges:=False

MsgBox "Data extraction and organization completed! Files saved to: " & mainFolder, vbInformation

Exit Sub

ErrorHandler:

MsgBox "An error occurred: " & Err.Description & vbCrLf & \_

"Please check the worksheet name or data format.", vbCritical

End Sub