```vba
'Function to input a sheet name and output true or false if the sheet exists
Function WorksheetExists(sName As String) As Boolean
    WorksheetExists = Evaluate("ISREF('" & sName & "'!A1)")
End Function

'Function to see if folder exists
Function FolderExists(FolderPath As String) As Boolean
    FolderExists = True
    On Error Resume Next
    ChDir FolderPath
    If Err <> 0 Then FolderExists = False
    On Error GoTo 0
End Function


Sub convert_sheet_to_table()
''''''''''''''''''''''''''''''''''''''''''''''''''''''''
'Sub function to copy a sheet in the range format and
'convert it to a table format
''''''''''''''''''''''''''''''''''''''''''''''''''''''''


'Define the variables to be used
Dim ws1 As Worksheet
Dim objTable As ListObject

'Test if the old "COMB" sheet was deleted. If it is not
'A message will pop up to delete the "COMB" sheet
'The user will have to click a button to finalize deletion
If WorksheetExists("COMB") = True Then
    MsgBox "Deleted old 'COMB' sheet"
    ActiveWindow.SelectedSheets.Delete
End If

'Sets the sheet "All" to a variable called ws1
Set ws1 = ThisWorkbook.Worksheets("All")

'Copies the ws1 variable and assigned it to a new sheet
ws1.Copy ThisWorkbook.Sheets(Sheets.count)

'Renames the new sheet to "COMB"
Sheets(ActiveSheet.Name).Name = "COMB"

'Deletes all activie queries before converting the sheet to the table.
'The is necessary otherwise an error occurs
ActiveSheet.QueryTables("IORS - Weekly Excel Export (P6)").Delete

'Convert the activie sheet which is "COMB" to a table format
ActiveSheet.ListObjects.Add(xlSrcRange, Range([A1].End(xlDown), [A1].End(xlToRight)), , xlYes).Name = "COMB"

End Sub

Sub group_by_departments()
```

```vba
    'Define variables
    Dim i, k, firstrow, lastrow, lrow, arraysize, total, start_index, end_index As Long
    Dim count As Long
    Dim dept_array(), sheet_name, indexs, old_fldrpath, new_filename, old_filename, path As String

    Dim HeaderRow, DataRows As Range

    Dim ws2 As Worksheet

    Dim CalcMode, ViewMode As Long

    Dim tdate As Date

    Dim fso, objFiles, obj, lngFileCount As Object

    'Need to set reference to 'Microsoft Scripting Runtime'in Tools -> References
    Dim dict As Scripting.Dictionary


    'Set application settings to speed up calculations
    With Application
        CalcMode = .Calculation
        .Calculation = xlCalculationManual
        .ScreenUpdating = False
        .DisplayAlerts = False
    End With

    'Select the current active sheet
    With ActiveSheet
        .Select
        ViewMode = ActiveWindow.View
        ActiveWindow.View = xlNormalView
        .DisplayPageBreaks = False


''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''

        'Check to see if the All sheet exists
        If WorksheetExists("All") <> True Then
            MsgBox "Error: Sheet named 'All' does not exist"
            Exit Sub
        End If

        'Error check to make sure All Sheet is selected
        If ActiveSheet.Name <> "All" Then
            MsgBox "Error: Sheet named 'All' is not currently selected"
            MsgBox "Current sheet will be deleted"
            ActiveWindow.SelectedSheets.Delete
            Exit Sub
        End If

        'Error check delete extra sheets before script runs
        If ThisWorkbook.Worksheets.count > 1 Then
            MsgBox "Error: Too many sheets exist"
            MsgBox "All sheets EXCEPT 'All' will be deleted"
```

```vba
      For Each ws2 In Application.ActiveWorkbook.Worksheets
         If ws2.Name <> "All" Then
            ws2.Delete
         End If
      Next
      MsgBox "Please click 'Start' again"
      Exit Sub
   End If


   path = ActiveWorkbook.path & "\Upload Files"
   'Error check to see if folder to save new files exists
   If FolderExists(path) <> True Then
      MsgBox "Error: The folder 'Upload Files' does not exist in current files directory"
      MsgBox "Please create folder or verify folder exists with name 'Upload Files' in current directory or contact
Administrator for help"
      Exit Sub
   End If


   'Error check to see if folder to archive files exists
   path = ActiveWorkbook.path & "\Archives"
   If FolderExists(path) <> True Then
      MsgBox "Error: The folder 'Archives' does not exist in current files directory"
      MsgBox "Please create folder or verify folder exists with name 'Archives' in current directory or contact
Administrator for help"
      Exit Sub
   End If




'''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''


   tdate = Now()

   'Delete all completed activities
   ActiveSheet.Range("A1").AutoFilter
   ActiveWorkbook.Worksheets("All").AutoFilter.Sort.SortFields.Clear
   ActiveWorkbook.Worksheets("All").AutoFilter.Sort.SortFields.Add Key:=Range( _
   "K:K"), SortOn:=xlSortOnValues, Order:=xlAscending, DataOption:= _
   xlSortNormal
 With ActiveWorkbook.Worksheets("All").AutoFilter.Sort
   .Header = xlYes
   .MatchCase = False
   .Orientation = xlTopToBottom
   .SortMethod = xlPinYin
   .Apply
 End With
   ActiveSheet.Range("A1").AutoFilter


   start_index = Range("K:K").Find(what:="Completed", after:=Range("K1")).Row
   end_index = Range("K:K").Find(what:="Completed", after:=Range("K1"), searchdirection:=xlPrevious).Row

   indexs = start_index & ":" & (end_index)
```

```vba
    sheet_name = "completed"
    Sheets.Add.Name = sheet_name

    Set HeaderRow = Worksheets("All").Range("1:1")
        HeaderRow.Copy
        Worksheets(sheet_name).Range(HeaderRow.Address).PasteSpecial Paste:=xlPasteColumnWidths, _
Operation:=xlNone, SkipBlanks:=False, Transpose:=False
        Worksheets(sheet_name).Paste

        Set DataRows = Worksheets("All").Range(indexs)
        DataRows.Copy
        Worksheets(sheet_name).Range("A1").Offset(1, 0).PasteSpecial Paste:=xlPasteColumnWidths, _
Operation:=xlNone, SkipBlanks:=False, Transpose:=False
        Worksheets(sheet_name).Paste

        ActiveSheet.Range("A1").AutoFilter

    Sheets("All").Select

        Rows(indexs).EntireRow.Delete

        'Sort by Dept
        ActiveSheet.Range("A1").AutoFilter
        ActiveWorkbook.Worksheets("All").AutoFilter.Sort.SortFields.Clear
        ActiveWorkbook.Worksheets("All").AutoFilter.Sort.SortFields.Add Key:=Range( _
        "D:D"), SortOn:=xlSortOnValues, Order:=xlAscending, DataOption:= _
        xlSortNormal
    With ActiveWorkbook.Worksheets("All").AutoFilter.Sort
        .Header = xlYes
        .MatchCase = False
        .Orientation = xlTopToBottom
        .SortMethod = xlPinYin
        .Apply
    End With
        ActiveSheet.Range("A1").AutoFilter

        'create array
        firstrow = .UsedRange.Cells(1).Row + 1
        lastrow = .UsedRange.Rows(.UsedRange.Rows.count).Row

        arraysize = lastrow
        ReDim dept_array(arraysize)

        'Debug.Print "Array Size: ", arraysize

        'input data into array
        i = 0
        For lrow = firstrow To lastrow Step 1
            If IsEmpty(Cells(lrow, 4).Value) Then
                dept_array(i) = ""
                i = i + 1
            Else
                dept_array(i) = .Cells(lrow, 4).Value
```

```vba
            i = i + 1
            'Debug.Print "Dept: ", .Cells(lrow, 4).Value
        End If
    Next lrow


    'input array into dictionary and count frequency of department
    Set dict = New Scripting.Dictionary
    k = 0
    Do While k <= arraysize
        If dict.Exists(dept_array(k)) Then
            dict.Item(dept_array(k)) = dict.Item(dept_array(k)) + 1
            total = total + 1
        Else
            dict.Add dept_array(k), 1
            total = total + 1
        End If
        k = k + 1
    Loop

'A method to uncomment to view dictionary
'For i = 0 To dict.count - 1
    'Debug.Print "Key:", dict.Keys(i), dict.Items(i)
'Next i


'Assign the row range of each department to a value in the dictionary
For i = 0 To dict.count - 1
    start_index = Range("D:D").Find(what:=dict.Keys(i), after:=Range("D1")).Row
    end_index = Range("D:D").Find(what:=dict.Keys(i), after:=Range("D1"), searchdirection:=xlPrevious).Row
    indexs = start_index & ":" & end_index
    dict.Item(i) = indexs
    Debug.Print "Key & Index:", dict.Keys(i), dict.Item(i)
Next i


'Sets the sheet "All" to a variable called ws1
Set ws2 = ThisWorkbook.Worksheets("All")

'
Set HeaderRow = Worksheets("All").Range("1:1")
HeaderRow.Copy

Set ws2 = ThisWorkbook.Worksheets("All")

k = (dict.count / 2) - 1
Debug.Print "Count:", k


For i = k To 0 Step -1
    If dict.Keys(i) = "" Then
        sheet_name = "no_dept"
        Sheets.Add.Name = sheet_name

        Set HeaderRow = Worksheets("All").Range("1:1")
```

```vba
        HeaderRow.Copy
        Worksheets(sheet_name).Range(HeaderRow.Address).PasteSpecial Paste:=xlPasteColumnWidths,
Operation:=xlNone, SkipBlanks:=False, Transpose:=False
        Worksheets(sheet_name).Paste

        Set DataRows = Worksheets("All").Range(dict.Item(i))
        DataRows.Copy
        Worksheets(sheet_name).Range("A1").Offset(1, 0).PasteSpecial Paste:=xlPasteColumnWidths,
Operation:=xlNone, SkipBlanks:=False, Transpose:=False
        Worksheets(sheet_name).Paste

        ActiveSheet.Range("A1").AutoFilter

      Else
        sheet_name = dict.Keys(i)
        Sheets.Add.Name = sheet_name

        Set HeaderRow = Worksheets("All").Range("1:1")
        HeaderRow.Copy
        Worksheets(sheet_name).Range(HeaderRow.Address).PasteSpecial Paste:=xlPasteColumnWidths,
Operation:=xlNone, SkipBlanks:=False, Transpose:=False
        Worksheets(sheet_name).Paste


        Set DataRows = Worksheets("All").Range(dict.Item(i))
        DataRows.Copy
        Worksheets(sheet_name).Range("A1").Offset(1, 0).PasteSpecial Paste:=xlPasteColumnWidths,
Operation:=xlNone, SkipBlanks:=False, Transpose:=False
        Worksheets(sheet_name).Paste

        ActiveSheet.Range("A1").AutoFilter

        'new_filename = sheet_name & " " & tdate

      End If

    Next i

End With


'''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''''

    'Ron DeBruin script to create a workbook from every worksheet in your workbook at
http://www.rondebruin.nl/win/s3/win007.htm
    'Working in 97-2013
    Dim FileExtStr As String
    Dim FileFormatNum As Long
    Dim Sourcewb As Workbook
    Dim Destwb As Workbook
    Dim sh As Worksheet
    Dim DateString As String
    Dim FolderName As String
```

```
With Application
   .ScreenUpdating = False
   .EnableEvents = False
   .Calculation = xlCalculationManual
End With


'Copy every sheet from the workbook with this macro
Set Sourcewb = ThisWorkbook



'Create new folder to save the new files in
DateString = Format(Now, "mm.dd.yyyy @ hh-mm AM/PM")
path1 = ActiveWorkbook.path & "\Upload Files\"
path2 = ActiveWorkbook.path & "\Archives\"
FolderName1 = path1
FolderName2 = path2

'Copy every visible sheet to a new workbook
For Each sh In Sourcewb.Worksheets

   'If the sheet is visible then copy it to a new workbook
   If sh.Visible = -1 Then
      sh.Copy

      'Set Destwb to the new workbook
      Set Destwb = ActiveWorkbook

      'Determine the Excel version and file extension/format
      With Destwb
         If Val(Application.Version) < 12 Then
            'You use Excel 97-2003
            FileExtStr = ".xls": FileFormatNum = -4143
         Else
            'You use Excel 2007-2013
            If Sourcewb.Name = .Name Then
               MsgBox "Your answer is NO in the security dialog"
               GoTo GoToNextSheet
            Else
               Select Case Sourcewb.FileFormat
               Case 51: FileExtStr = ".xlsx": FileFormatNum = 51
               Case 52:
                  If .HasVBProject Then
                     FileExtStr = ".xlsm": FileFormatNum = 52
                  Else
                     FileExtStr = ".xlsx": FileFormatNum = 51
                  End If
               Case 56: FileExtStr = ".xls": FileFormatNum = 56
               Case Else: FileExtStr = ".xlsb": FileFormatNum = 50
               End Select
            End If
         End If
      End With

      'Change all cells in the worksheet to values if you want
```

```vba
         If Destwb.Sheets(1).ProtectContents = False Then
            With Destwb.Sheets(1).UsedRange
               .Cells.Copy
               .Cells.PasteSpecial xlPasteValues
               .Cells(1).Select
            End With
            Application.CutCopyMode = False
         End If


        'Save the new workbook and close it
        With Destwb
           .SaveAs FolderName1 _
               & "IORS - Weekly Excel Export" & " " & "(" & Destwb.Sheets(1).Name & ")" & " " & FileExtStr, _
                FileFormat:=FileFormatNum

           .SaveAs FolderName2 _
               & "IORS - Weekly Excel Export" & " " & "(" & Destwb.Sheets(1).Name & ")" & " " & DateString &
FileExtStr, _
                 FileFormat:=FileFormatNum
           .Close False

        End With
      End If
GoToNextSheet:
   Next sh

   MsgBox "You can find the files for Sharepoint in the folder named " & FolderName1 & " and the files each time this
report is ran in the folder " & FolderName2

   With Application
      .ScreenUpdating = True
      .EnableEvents = True
      .Calculation = xlCalculationAutomatic
   End With


'Turn back application settings
'With Application
      '.Calculation = CalcMode
      '.ScreenUpdating = True
'End With

End Sub
```