Sub DrillAcross()

Dim currentRowInProcess(20) 'pointer to current row in each process

Dim currentLowestSort(20) 'value of current lowest sort

Dim factIndexes(20) ' for each process, column index in results where facts begin

Worksheets("DrillAcrossResult").Cells.ClearContents

numRowHeaderCols = Worksheets("StartHere").Cells(9, 4).Value

numProcesses = Worksheets("StartHere").Cells(8, 4).Value

For i = 1 To numProcesses 'sorry for the ugly code here. I couldnt figure out how to assemble a text string and then execute it.

Select Case numRowHeaderCols

Case 1

Worksheets("Process" + Format(i)).Cells.Sort Key1:=Worksheets("Process" + Format(i)).Range("A1"), Order1:=xlAscending, Header:=xlYes

Case 2

Worksheets("Process" + Format(i)).Cells.Sort Key1:=Worksheets("Process" + Format(i)).Range("A1"), Order1:=xlAscending, Key2:=Worksheets("Process" + Format(i)).Range("B1"), Order2:=xlAscending, Header:=xlYes

Case 3

Worksheets("Process" + Format(i)).Cells.Sort Key1:=Worksheets("Process" + Format(i)).Range("A1"), Order1:=xlAscending, Key2:=Worksheets("Process" + Format(i)).Range("B1"), Order2:=xlAscending, Key3:=Worksheets("Process" + Format(i)).Range("C1"), Order3:=xlAscending, Header:=xlYes

Case 4

Worksheets("Process" + Format(i)).Cells.Sort Key1:=Worksheets("Process" + Format(i)).Range("A1"), Order1:=xlAscending, Key2:=Worksheets("Process" + Format(i)).Range("B1"), Order2:=xlAscending, Key3:=Worksheets("Process" + Format(i)).Range("C1"), Order3:=xlAscending, Key4:=Worksheets("Process" + Format(i)).Range("D1"), Order4:=xlAscending, Header:=xlYes

Case 5

Worksheets("Process" + Format(i)).Cells.Sort Key1:=Worksheets("Process" + Format(i)).Range("A1"), Order1:=xlAscending, Key2:=Worksheets("Process" + Format(i)).Range("B1"), Order2:=xlAscending, Key3:=Worksheets("Process" + Format(i)).Range("C1"), Order3:=xlAscending, Key4:=Worksheets("Process" + Format(i)).Range("D1"), Order4:=xlAscending, Key5:=Worksheets("Process" + Format(i)).Range("E1"), Order5:=xlAscending, Header:=xlYes

End Select

Next i

CurrentOutputRow = 1

For j = 1 To numRowHeaderCols

Worksheets("DrillAcrossResult").Cells(1, j).Value = Worksheets("Process1").Cells(1, j).Value ' copy row headers from Process 1 to row 1 of Result

Next j

factcolindex = numRowHeaderCols + 1 'start marching across results fact columns, adding headings from each process n next block

For i = 1 To numProcesses

j = numRowHeaderCols + 1

lookformorefacts:

factcolheader = Worksheets("Process" + Format(i)).Cells(1, j).Value

If factcolheader <> "" Then

Worksheets("DrillAcrossResult").Cells(1, factcolindex).Value = factcolheader

If j = numRowHeaderCols + 1 Then

factIndexes(i) = factcolindex ' establishes column index in results where this process's facts start

End If

factcolindex = factcolindex + 1

j = j + 1

GoTo lookformorefacts

End If

currentRowInProcess(i) = 2 'initialize row pointer in each process to first data row, assumes heading rows present

If Worksheets("Process" + Format(i)).Cells(2, 1).Value = "" Then 'looks in what should be the first row of the answer set

currentRowInProcess(i) = 0 'protects against an empty answer set

End If

Next i

CurrentOutputRow = 2

' search all processes to find the lowest sort valued row headers

' skip processes where currentRowInProcess is zero (processes where all rows have been read)

' stop when all currentRowInProcess's are zero

' output all processes matching these lowest valued row headers

' increment currentRowInProcess for each

' set to zero when process has been completely read

' loop to beginning

grandloop:

For i = 1 To numRowHeaderCols

currentLowestSort(i) = "ZZZZZZ" 'initialize currentLowestSort each time we search for lowest row header values

Next i

finishedAllProcesses = True 'set to False below if find any remaining rows in any process

For i = 1 To numProcesses

For j = 1 To numRowHeaderCols

If currentRowInProcess(i) = 0 Then

GoTo nextProcess 'we have reached the end of this process, skip it

End If

finishedAllProcesses = False

testval = Worksheets("Process" + Format(i)).Cells(currentRowInProcess(i), j) 'look at header col value in a particular process

If testval < currentLowestSort(j) Then 'test if header col val is a minimum

currentLowestSort(j) = testval

For k = j + 1 To numRowHeaderCols

currentLowestSort(k) = "ZZZZZZ" 'reinitialize the rest of the sort values

Next k

End If

Next j

nextProcess:

Next i

If finishedAllProcesses Then

MsgBox "Report Finished!"

Exit Sub

End If

' we now have the currentLowestSort

' scan all the processes with these row header values, output them, and increment their currentRowInProcess values

For i = 1 To numRowHeaderCols

Worksheets("DrillAcrossResult").Cells(CurrentOutputRow, i).Value = currentLowestSort(i) ' populate this row's headers

Next i

For i = 1 To numProcesses 'check each process to see if current row equals lowest sort

foundlowestrowheaders = False

If currentRowInProcess(i) = 0 Then

GoTo keeplookingatrowheaders

End If

For j = 1 To numRowHeaderCols

If Worksheets("Process" + Format(i)).Cells(currentRowInProcess(i), j).Value = currentLowestSort(j) Then

foundlowestrowheaders = True

If j < numRowHeaderCols Then

GoTo keeplookingatrowheaders

Else

Exit For

End If

Else

foundlowestrowheaders = False

Exit For

End If

keeplookingatrowheaders:

Next j

If foundlowestrowheaders Then

j = numRowHeaderCols + 1 'column containing fact

k = 0 ' count of columns added from this process

outputanothercolumn:

outputval = Worksheets("Process" + Format(i)).Cells(currentRowInProcess(i), j).Value

If outputval <> "" Then

Worksheets("DrillAcrossResult").Cells(CurrentOutputRow, factIndexes(i) + k).Value = Worksheets("Process" + Format(i)).Cells(currentRowInProcess(i), j).Value

j = j + 1

k = k + 1

GoTo outputanothercolumn

End If

currentRowInProcess(i) = currentRowInProcess(i) + 1

If Worksheets("Process" + Format(i)).Cells(currentRowInProcess(i), 1).Value = "" Then ' test if finished with this process

currentRowInProcess(i) = 0

End If

End If

Next i

CurrentOutputRow = CurrentOutputRow + 1

GoTo grandloop

End Sub

© 2013 Kimball Group. All rights reserved.