

Re: Range Object Misunderstanding

Source:

<http://www.tech-archive.net/Archive/Excel/microsoft.public.excel.programming/2005-08/msg02978.html>

- *From:* Ken McLennan <kenrmcl@xxxxxxxxxxxxx>
 - *Date:* Mon, 8 Aug 2005 21:59:02 +1000
-

G'day there Bob,

> Do you declare all of your variable up-front? Where is gStrArray and rngSrt
> declared? What and how is range Groups defined?

Sorry about that. I should have posted more information. That's what happens when you try to rush things.

Last one first, I have a worksheet titled "Data". This worksheet is a Worksheet Object called "dSht" (The sheet named "Calendar" is an object called "cSht"; "Overtime" is "oSht"; etc). On "dSht" I have a named range "Groups", defined by "=OFFSET(Data!\$J\$1,0,0,COUNTA(Data!\$J:\$J),1)" in my Names dialogue.

gStrArray is declared in a general module called "MainModule":

```
Public gStrArray() As String
```

rngSrt is a function in the same module:

```
Public Function rngSrt(List() As String, UpDown As Boolean)
```

```
' Generic array sorting routine  
' "List" is string array for sorting  
' "UpDown" is direction - True = ascending  
' "BubbleSort" coding courtesy of John Walkenbach
```

```
Dim First As Integer, Last As Integer  
Dim i As Integer, j As Integer  
Dim gStr1 As String  
First = LBound(List)  
Last = UBound(List)
```

```
For i = First To Last - 1  
For j = i + 1 To Last  
If UpDown = True Then  
If UCase(List(i)) > UCase(List(j)) Then  
gStr1 = List(j)
```

Re: Range Object Misunderstanding

```
List(j) = List(i)
List(i) = gStr1
End If
Else
If UCase(List(i)) < UCase(List(j)) Then
gStr1 = List(j)
List(j) = List(i)
List(i) = gStr1
End If
End If
Next j
Next i

' Return sorted array to calling routine
rngSrt = List

End Function
```

The idea is that the ListBox on my userform allows me to add or remove entries from the named range "Groups". The entries are loaded into my string array "gStrArray()", then the listbox displayed from those entries. Whatever is in gStrArray() is put back into "Groups" when the userform is terminated. Well, that's the theory anyway.

Hmmm... I've just realised that I've moved away from what I intended. I originally used this sort of array manipulation so that I'd have a copy of the starting entries from q