appendix

C

Source Code

Listed are a number of functions that I have found useful for modeling that you can copy and use. These functions will be beneficial to have in your function library. I have provided a list of these functions and a short brief on how to enter them. Before working with these functions, be sure to read Chapter 18, particularly the section on using the Visual Basic Editor (VBE).

## Find Saturday

The key feature of this function is not that it finds Saturday but that you can change the weekday to any day that you choose. I use this function when I need a specific weekday to trigger some event.

Here's the code:

Public FunctionFindSaturday(InputDate As Date)

' Returns the date of the first Saturday following the Inputdate

   FindSaturday = FormatDateTime(InputDate + (7 - Weekday(InputDate)))

End Function

## Remove Spaces

Here's the code:

Public Function RemoveSpaces(strInput As String)

' Removes all spaces from a string of text

Test:

   If InStr(strInput, " ") = 0 Then

      RemoveSpaces = strInput

   Else

      strInput = Left(strInput, InStr(strInput, " ") - 1) \_

      & Right(strInput, Len(strInput) - InStr(strInput, " "))

      GoTo Test

   End If

End Function

## Sum Colors

Function SumColor(rColor As Range, rSumRange As Range)

'Sums cells based on a specified fill color.

Dim rCell As Range

Dim iCol As Integer

Dim vResult

iCol = rColor.Interior.ColorIndex

For Each rCell In rSumRange

If rCell.Interior.ColorIndex = iCol Then

vResult = WorksheetFunction.Sum(rCell) + vResult

End If

Next rCell

SumColor = vResult

End Function

## Count Colors

Function CountColor(rColor As Range, rSumRange As Range)

'Counts cells based on a specified fill color

Dim rCell As Range

Dim iCol As Integer

Dim vResult

iCol = rColor.Interior.ColorIndex

For Each rCell In rSumRange

If rCell.Interior.ColorIndex = iCol Then

vResult = vResult + 1

End If

Next rCell

CountColor = vResult

End Function

## Sort Using Colors

Here's the code:

Function ColorRank(ColorOrder As Range, LookCell As Range)

Dim i As Integer

Dim ICol1 As Integer

Dim ICol2 As Integer

i = 1

ICol2 = -1

ColorRank = 0

'Loop until match is found

Do Until ICol1 = ICol2

'Replace "Interior" with Font to sort by Font color

ICol1 = ColorOrder(i, 1).Interior.ColorIndex

ICol2 = LookCell.Interior.ColorIndex

If i = ColorOrder.Rows.Count + 1 Then

'No Match found place in Text

ColorRank = "No color match!!!"

Exit Do End If

'Pass the Row number of the color match

ColorRank = i

i = i + 1

Loop

End Function

## Sum the Bottom and Top Numbers in a Range

Here's the code:

Function SumTopBottomN(rRange As Range, N As Long, Optional bBottomN As Boolean) As Single

Dim strAddress As String

On Error Resume Next

strAddress = rRange.Address

If bBottomN = False Then

SumTopBottomN = Evaluate("=SUMPRODUCT((" \_

& strAddress & ">=LARGE(" & strAddress & "," & X & "))\*(" & strAddress & "))")

Else

SumTopBottomN = Evaluate("=SUMPRODUCT((" \_

& strAddress & "<=SMALL(" & strAddress & "," & X & "))\*(" & strAddress & "))")

End If

End Function

You will use this formula in one of two ways.

=SumTopBottomN($A$2:$A$100,10) For top 10

OR

=SumTopBottomN($A$2:$A$100,10,TRUE) For bottom 10

## Remove Formula

Here's the code:

Sub Remove\_All\_Formulas\_In\_All\_Sheets 'No\_formula\_Macro()

'Striping a workbook of formulas

Dim SH As Worksheet

For Each SH In Worksheets

SH.Activate

Application.ScreenUpdating = False

Cells.Select

Selection.Copy

Selection.PasteSpecial Paste:=xlValues, Operation:=xlNone, \_

SkipBlanks:= False, Transpose:=False

Next

End Sub

## Replace External Links with Their Values

Here's the code:

Sub Tester3()

Dim cell As Range

For Each cell In ActiveSheet.UsedRange.SpecialCells(xlFormulas)

If InStr(cell.Formula, "[") Then

cell.Formula = cell.Value

End If

Next

End Sub

## Collect Statistics on the Worksheet

Here's the code:

Function SheetStats()

'Note in XL97 and above xlcelltypeconstants and xlcelltypeformula

Set rng1 = ActiveSheet.UsedRange

On Error Resume Next

numConstants = rng1.SpecialCells(xlConstants).Count

If Err <> 0 Then numConstants = 0: Err = 0

numerrors = rng1.SpecialCells(xlConstants, xlErrors).Count

If Err <> 0 Then numerrors = 0: Err = 0

numLogical = rng1.SpecialCells(xlConstants, xlLogical).Count

If Err <> 0 Then numLogical = 0: Err = 0

numText = rng1.SpecialCells(xlConstants, xlTextValues).Count

If Err <> 0 Then numText = 0: Err = 0

numNumbers = rng1.SpecialCells(xlConstants, xlNumbers).Count

If Err <> 0 Then numNumbers = 0: Err = 0

numformulas = rng1.SpecialCells(xlFormulas).Count

If Err <> 0 Then numformulas = 0: Err = 0

numBlanks = rng1.SpecialCells(xlBlanks).Count

If Err <> 0 Then numBlanks = 0: Err = 0

Msg = "Address: " & Chr(9) & rng1.Address & Chr(10) & \_

"Last Row: " & Chr(9) & rng1.Rows(rng1.Rows.Count).Row & Chr(10) & \_

"Last Column: " & Chr(9) & rng1.Columns(rng1.Columns.Count).Column & Chr(10) & \_

"Total Cells: " & Chr(9) & rng1.Count & Chr(10) & \_

" Formulas: " & Chr(9) & numformulas & Chr(10) & \_

" Blanks: " & Chr(9) & numBlanks & Chr(10) & \_

" Constants:" & Chr(9) & numConstants & Chr(10)

Mg2 = "Errors: " & Chr(9) & numerrors & Chr(10) & \_

"Logical: " & Chr(9) & numLogical & Chr(10) & \_

"Text: " & Chr(9) & numText & Chr(10) & \_

"Numbers: " & Chr(9) & numNumbers

title1 = "SheetStats for " & Application.ActiveSheet.Name & \_

" in " & Application.ActiveWorkbook.FullName

iANS = MsgBox(Msg & Mg2, , title1)

End Function

## Look Up Information in All Sheets in an Excel Workbook

Here's the code:

Function VLOOKAllSheets(Look\_Value As Variant, Tble\_Array As Range, \_

Col\_num as Integer, Optional Range\_look as Boolean)

'Use VLOOKUP to Look across ALL Worksheets and stops at the first match found.

Dim wSheet As Worksheet

Dim vFound

On Error Resume Next

For Each wSheet In ActiveWorkbook.Worksheets

With wSheet

Set Tble\_Array = .Range(Tble\_Array.Address)

vFound = WorksheetFunction.VLookup \_

(Look\_Value, Tble\_Array, \_

Col\_num, Range\_look)

End With

If Not IsEmpty(vFound) Then Exit For

Next wSheet

Set Tble\_Array = Nothing

VLOOKAllSheets = vFound

End Function