

SplitSheet and SplitBook Functions

1 SplitSheet Function

Once gaps were identified and missing documents or new revisions updated on both systems; the data needed to be collated accordingly for reporting. When reporting to different recipients that were interested only in certain parts of data, it was necessary to split the data. Document owners were interested in status of their documents only; therefore data was split by owner.

VBA Code 'AddSheets()' (Figure 1) creates multiple worksheets by values listed in one column (i.e. 'Owners' in columns A of both worksheets). It splits the data of one master worksheet into separate worksheets per document owner (values of first column in this example) into multiple worksheets without having to manually sort it and copy-paste it from one worksheet to another. It also keeps the master worksheet for further editing or filtering if needed.

To split the master worksheet into multiple worksheets by values of first column:

1. Press **Alt+F11** to open Visual Basic Editor (VBE);
2. On the Top Menu Select Insert ► Module from the context menu;
3. Copy and paste VBA code to Code window;
4. Close VBE window;
5. Press Alt+F8 to open the 'Macro' dialog;
6. Select the '**AddSheets()**' macro and click 'Run'.



```

Sub AddSheets()
Dim s$, r
Application.ScreenUpdating = False
With Sheets("BC").Range("A1").CurrentRegion
.Parent.AutoFilterMode = False
For Each r In .Offset(1).Resize(.Rows.Count - 1).Columns(1).Value
If InStr(s, r) = 0 Then
If Not Evaluate("ISREF("'" & r & "'!A1)") Then
Sheets.Add(after:=Sheets(Sheets.Count)).Name = r
Else
Sheets(r).UsedRange.ClearContents
End If
.AutoFilter 1, r
.Copy Sheets(r).Range("A1")
s = s & r
End If
Next
.AutoFilter
End With
Application.ScreenUpdating = True
End Sub

```

Figure 1 - 'AddSheets()' - VBA code for creating worksheets from selected cells



Change highlighted worksheet name (BC) on VBA code to the title of worksheet that needs to be split.

2 SplitBook Function

VBA Code 'Splitbook()' (Figure 2) automatically saves all worksheets as separate workbooks titled by worksheet names. This assured we distributed only significant information per recipient and eliminated the need for each document owner to search for the relevant tab in a big workbook.

To divide a workbook with multiple sheets and automatically save each worksheet separately:

1. Press **Alt+F11** to open Visual Basic Editor (VBE);
2. On the Top Menu Select Insert ► Module from the context menu;
3. Copy and paste VBA code to Code window ;
4. Close VBE window;
5. Press Alt+F8 to open the 'Macro' dialog;
6. Select the '**Splitbook()**' macro and click 'Run'.



```

Sub Splitbook()
Dim xPath As String
xPath = Application.ActiveWorkbook.Path
Application.ScreenUpdating = False
Application.DisplayAlerts = False
For Each xWs In ThisWorkbook.Sheets
    xWs.Copy
    Application.ActiveWorkbook.SaveAs Filename:=xPath & "\" & xWs.Name &
    ".xlsx"
    Application.ActiveWorkbook.Close False
Next
Application.DisplayAlerts = True
Application.ScreenUpdating = True
End Sub

```

Figure 2 - Splitbook()- VBA code for saving all worksheets of one workbook as separate files

Lessons Learned

Working with Big Data can be confusing and time-consuming task for inexperienced MS Excel users.

Using the Splitsheet and Splitbook functions to circulate reports maintains the integrity of the data and keeps it appropriate for each audience. It eliminates the need of filtering and/or deleting irrelevant data which can lead to missing important information and reduces the file size for each recipient.