

VBA SOURCE CODE BOOK

LABEL & NAME CARD CREATOR PT. 1

Label Text	Data	Show Label	Show Data
ID #	1001		
Last Name	Aames	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
First Name	Lisa	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Gender	Female	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Address	7167 Burning Circles	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
City	Sunrise Vista	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
State	CA	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Zip	90842	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Person Type	Owner	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Company	Active	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Phone	514284049	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Other Phone	794286992	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Email	pe@TestCompany.com	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Excel For Freelancers

Learn How To Create &
Customize Unlimited Labels &
Name Cards In Excel Without
Mail Merge



[DOWNLOAD
APPLICATION](#)



[WATCH ENTIRE
SERIES](#)



[VIEW
TRAINING](#)

by: *Randy Austin*

ABOUT THE AUTHOR

A two-time Microsoft MVP & lifetime Excel enthusiast, Randy Austin founded Excel For Freelancers in 2017. Excel For Freelancers quickly became the most prominent resource Excel for developers to learn how to turn their passion for Excel into profits by building & selling their own excel-based applications for passive & recurring income.

With nearly 300,000 YouTube subscribers, 14,000,000 video views, 200+ comprehensive training videos, and a thriving 40,000 member Facebook community, Excel For Freelancers has positioned itself as the #1 Excel developers resource in the world.

Get free content, training, and downloads just by clicking any of the free resources below:



[WEBSITE](#)



[YOUTUBE](#)



[FACEBOOK](#)



[TWITTER](#)



[INSTAGRAM](#)



[TELEGRAM](#)



[RUMBLE](#)



Microsoft®
Most Valuable
Professional



OUR COURSES & PRODUCTS



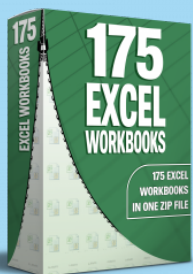
This comprehensive program will take you through a 12-phase process that will turn your enthusiasm for Excel into passive income.

[Click here to learn more](#)



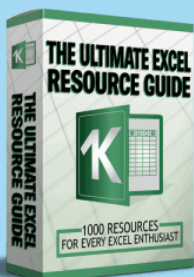
16 hour masterclass that will teach you the tips, tricks and techniques on how to create a dynamic single-click dashboard, and a ton more

[Click here to learn more](#)



Incredible Package of 175 of my BEST Applications into a SINGLE ZIP File which also includes the "175 Workbook Library".

[Click here to learn more](#)



With 1000 live links, continuously updating content, sort-able and filterable items, you will always have exactly what you need, when you need it.

[Click here to learn more](#)

Projects.....	2
VBAProject	2
Documents	2
Sheet1.....	2
(Declarations).....	2
Worksheet_Change [Sub].....	2
Worksheet_SelectionChange [Sub]	2
Sheet2.....	5
(Declarations).....	5
Sheet3.....	7
(Declarations).....	7
Sheet4.....	9
(Declarations).....	9
ThisWorkbook	11
(Declarations).....	11
Modules.....	13
CodeReset_Macs	13
(Declarations).....	13
CalculateRunTime_Minutes [Sub]	13
ResetCalc [Sub]	13
StartClock [Sub]	13
StopCalc [Sub]	13
StopClock [Sub]	13
CreateLabelMacs	15
(Declarations).....	15
BuildLabelSheet [Sub].....	15
FilterDataCreateLabels [Sub]	16
PrintLabels [Sub].....	18
DataWkBkMacros	20
(Declarations).....	20
BrowseForData [Sub]	20
GetDataDetails [Sub]	20
FilterBtnMacros	24
(Declarations).....	24
AddFilt_Numb_GreaterThan [Sub]	24
AddFilt_Numb_LessThan [Sub].....	24
AddFilt_Text_Contains [Sub].....	24
AddFilt_Text_EndsWith [Sub]	24
AddFilt_Text_StartsWith [Sub]	24
ShowFilterGroup [Sub]	25
Label_Load	27
(Declarations).....	27
LoadLabel [Sub]	27
LabelDataMacros	29
(Declarations).....	29
ImportLabelData [Sub]	29
LabelDesignMacs	32
(Declarations).....	32
GroupTemplateShapes [Sub]	32
UngroupTemplateShape [Sub]	32
TabMacros	34
(Declarations).....	34
CreateLabelStep1 [Sub]	34
CreateLabelStep2 [Sub]	34
CreateLabelStep3 [Sub]	34
CreateLabelStep4 [Sub]	34
ResetHideSteps [Sub].....	35

Option Explicit

```

1 Private Sub Worksheet_Change(ByVal Target As Range)
2
3     If Not Intersect(Target, Range("E8" )) Is Nothing Then
4         Range("B4" ).Value = True 'Set Template Change to True
5         LoadLabel 'Load Label details on change of label
6     End If
7
8     If Not Intersect(Target, Range("I21" )) Is Nothing Then
9         If Cells(Target.Row, Target.Column).Value <> Empty Then GetDataDetails 'Get
10        Destinationworksheet data on worksheet change
11    End If
12
13    'On Change of Column Format, when Column D is not empty, and not on Mapping load, then
14    Update Format Validation and Value
15    If Not Intersect(Target, Range("H27:H77" )) Is Nothing And Range("D" & Target.Row).Value
16    <> Empty Then
17        Range("I" & Target.Row).Validation.Delete
18        If Range("H" & Target.Row).Value <> "General" Then Range("I" & Target.Row).
19        Validation.Add Type:=xlValidateList, AlertStyle:=xlValidAlertStop, Operator:=xlBetween,
20        Formula1:="=" & Range("H" & Target.Row).Value & "FormatList"
21    End If
22
23    'On Change Of Format, update reformat sample Data
24    If Not Intersect(Target, Range("I27:I77" )) Is Nothing And Range("I" & Target.Row).Value
25    <> Empty Then
26        Range("E" & Target.Row).NumberFormat = Range("I" & Target.Row).Value 'Update Format
27        for sample data
28    End If
29 End Sub

```

```

30 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
31     If Target.CountLarge > 1 Then Exit Sub
32     Dim DesignRow As Long, DesignColumn As Long
33     Dim ShapeName As String, ShapeFormula As String
34     If Target.CountLarge > 5 Then Exit Sub
35     'Display Filter Options on Selection
36     If Not Intersect(Target, Range("G27:G77" )) Is Nothing And Range("D" & Target.Row).Value
37     <> Empty And Target.Count < 2 Then
38         Range("B1" ).Value = Target.Row
39         ShowFilterGroup 'Display Text Filter Buttons
40     Else: 'Other than Filter Selection, hide filterGroup
41         Shapes("FiltGrp" ).Visible = msoFalse
42     End If
43
44     'Check/Uncheck Data & Label Options
45     If Not Intersect(Target, Range("F83:G132" )) Is Nothing And Range("D" & Target.Row).Value
46     <> Empty Then
47         DesignColumn = Target.Column 'Selected Design Column
48         DesignRow = Target.Row 'Selected Design Row
49         If DesignColumn = 6 Then ShapeName = "Label" & DesignRow Else ShapeName = "Data" &
50         DesignRow
51         If DesignColumn = 6 Then ShapeFormula = "=$D$" & DesignRow Else ShapeFormula = "=$E$"
52         & DesignRow
53         If Target.Value = "ü" Then
54             Target.ClearContents
55             On Error Resume Next
56             Shapes(ShapeName).Delete 'Delete shape
57             On Error GoTo 0
58         Else:
59             Target.Value = "ü"
60             'Create & Hide Label Shapes

```

1 2 3

```
51     1 2 3 Shapes("SampleTextBox" ).Copy
52     Range("H83" ).Select
53     Paste
54     Selection.Name = ShapeName
55     With Shapes(ShapeName)
56     .Left = Sheet1.Range("H83" ).Left
57     .Top = Sheet1.Range("H83" ).Top
58     .IncrementLeft 15
59     .IncrementTop 10
60     .DrawingObject.Formula = ShapeFormula
61     .Placement = xlFreeFloating
62     .Visible = msoCTrue
63     End With
64     End If
65     Range("E" & Target.Row).Select 'Select out of the cell
66     End If
67     End Sub
68
```

A

Add, 2
AlertStyle, 2

C

Cells, 2
ClearContents, 2
Column, 2
Copy, 3
Count, 2
CountLarge, 2

D

Delete, 2
DesignColumn, 2
DesignRow, 2
DrawingObject, 3

E

Empty, 2
Explicit, 2

F

Formula, 3
Formula1, 2

G

GetDataDetails, 2

I

IncrementLeft, 3
IncrementTop, 3
Intersect, 2

L

Left, 3
LoadLabel, 2

M

msoCTrue, 3
msoFalse, 2

N

Name, 3
NumberFormat, 2

O

Operator, 2

P

Paste, 3
Placement, 3

R

Range, 2, 3
Row, 2, 3

S

Selection, 3
ShapeFormula, 2, 3
ShapeName, 2, 3
Shapes, 2, 3
Sheet1, 3
ShowFilterGroup, 2

T

Target, 2, 3
Top, 3

V

Validation, 2
Value, 2
Visible, 2, 3

W

Worksheet_Change, 2
Worksheet_SelectionChange, 2

X

xlBetween, 2
xlFreeFloating, 3
xlValidAlertStop, 2
xlValidateList, 2

1 Option Explicit

2

E

Explicit, 5

1 Option Explicit

2

E

Explicit, 7

1 Option Explicit

2

E

Explicit, 9

1	Option Explicit
2	

E

Explicit, [11](#)

```
1 Option Explicit
2 Dim StartTime As Double
3 Dim SecondsElapsed As Double
4
5 Sub CalculateRunTime_Minutes()
6 'PURPOSE: Determine how many minutes it took for code to completely run
7 'SOURCE: www.TheSpreadsheetGuru.com/the-code-vault
8
9 Dim StartTime As Double
10 Dim MinutesElapsed As String
11
12 'Remember time when macro starts
13 StartTime = Timer
14
15 '*****
16 'Insert Your Code Here...
17 '*****
18
19 'Determine how many seconds code took to run
20 MinutesElapsed = Format((Timer - StartTime) / 86400, "hh:mm:ss" )
21
22 'Notify user in seconds
23 MsgBox "This code ran successfully in " & MinutesElapsed & " minutes" , vbInformation
24
25 End Sub
26
27 Sub ResetCalc()
28 With Application
29 .EnableEvents = True
30 .Calculation = xlCalculationAutomatic
31 ' .ScreenUpdating = True
32 End With
33 End Sub
34
35
36
37 Sub StartClock()
38
39 'Remember time when macro starts
40 StartTime = Timer
41 End Sub
42
43 Sub StopCalc()
44 With Application
45 .EnableEvents = False
46 .Calculation = xlCalculationManual
47 ' .ScreenUpdating = False
48 End With
49 End Sub
50
51 Sub StopClock()
52
53 'Determine how many seconds code took to run
54 SecondsElapsed = Round(Timer - StartTime, 2)
55
56 'Notify user in seconds
57 MsgBox "This code ran successfully in " & SecondsElapsed & " seconds" , vbInformation
58 End Sub
```


A

Application, 13

C

CalculateRunTime_Minutes, 13

Calculation, 13

E

EnableEvents, 13

Explicit, 13

F

Format, 13

M

MinutesElapsed, 13

MsgBox, 13

R

ResetCalc, 13

Round, 13

S

SecondsElapsed, 13

StartClock, 13

StartTime, 13

StopCalc, 13

StopClock, 13

T

Timer, 13

V

vbInformation, 13

X

xlCalculationAutomatic, 13

xlCalculationManual, 13

```

1 Option Explicit
2 Sub BuildLabelSheet()
3   If Sheet1.Range("B4").Value = False Then Exit Sub 'No need to rebuild label sheet unless
   template has changed
4   StopCalc
5   Dim LabSheetQty As Long, LabelCol As Long, LabelRow As Long, SetCount As Long
6   Dim LastLabelRow As Long, LastLabelCol As Long, SetCol As Long, SetRow As Long
7   Dim DataRow As Long, LastDataRow, MaxLabels As Long, MaxRows As Long
8   Dim LabHeight As Double, LabWidth As Double, ColumnSpace As Double, RowSpace As Double
9   Dim LeftMargin As Double, RightMargin As Double, TopMargin As Double, BottomMargin As
   Double
10  Dim PrintArea As Range
11  With Sheet1
12    LabSheetQty = .Range("E9").Value 'Label Sheet Qty
13    LabelCol = .Range("E11").Value 'Number of label columns
14    LabelRow = .Range("G11").Value 'Number of label rows
15    LabHeight = .Range("E10").Value 'Label Height
16    LabWidth = .Range("G10").Value 'Label Width
17    ColumnSpace = .Range("E13").Value 'Column Spacing
18    RowSpace = .Range("G13").Value 'Row Spacing
19    LeftMargin = .Range("E14").Value 'Left Margin Spacing
20    RightMargin = .Range("G14").Value 'Right Margin
21    TopMargin = .Range("E15").Value 'Top Margin
22    BottomMargin = .Range("G15").Value 'Bottom Margin
23    MaxLabels = .Range("H22").Value - 1 'Maximum # of Labels (without any filter)
24    MaxRows = MaxLabels / LabelCol
25    'Determine the Total # of Rows (Full Sheets) Round Up To Whole # (Maximum Rows / Rows
   Per sheet) * Number OF Rows Per Sheet
26    LastLabelRow = Application.WorksheetFunction.RoundUp((MaxRows / LabelRow), 0) *
   LabelRow
27    LastLabelCol = LabelCol + (LabelCol - 1)
28    LastLabelRow = LastLabelRow + (LastLabelRow - 1) 'Set the Last Label Row (Use full
   Sheets)
29  End With
30  With Sheet3 'Label Template
31    'Setup Columns
32    For SetCol = 1 To LastLabelCol Step 2
33      For SetCount = 1 To 3 'Run 3 times to set width
34        .Cells(1, SetCol).ColumnWidth = Application.InchesToPoints(LabWidth) * (.Cells(1,
   SetCol).ColumnWidth / .Cells(1, SetCol).Width) - 3
35      Next SetCount
36      If SetCol <> LastLabelCol Then .Cells(1, SetCol + 1).ColumnWidth = ColumnSpace
37    Next SetCol
38    'Setup Rows
39    For SetRow = 1 To LastLabelRow Step 2
40      .Cells(SetRow, 1).RowHeight = Application.InchesToPoints(LabHeight) 'Set Row Height
41      .Cells(SetRow + 1, 1).RowHeight = RowSpace
42    Next SetRow
43    With .PageSetup
44      .PrintArea = Range(Sheet3.Cells(1, 1), Sheet3.Cells(LastLabelRow, LastLabelCol)).
   Address
45      .LeftMargin = Application.InchesToPoints(LeftMargin)
46      .RightMargin = Application.InchesToPoints(RightMargin)
47      .TopMargin = Application.InchesToPoints(TopMargin)
48      .BottomMargin = Application.InchesToPoints(BottomMargin)
49      .HeaderMargin = 0
50      .FooterMargin = 0
51    End With
52    Sheet1.Range("B4").Value = False 'Reset Label Change to False
53  End With

```

```

1
54     ResetCalc
55 End Sub
56 Sub FilterDataCreateLabels()
57     Dim FirstFiltCol As Long, LastFiltCol As Long, TotalDataCol As Long
58     Dim LabelCol As Long, LabelRow As Long, LabelNumber
59     Dim TotalLabelCol As Long, TotalLabelRow As Long
60     Dim FirstLabelRow As Long, LastLabelRow As Long, LabelDataRow As Long
61     Dim FirstLabelCol As Long, LastLabelCol As Long, LabelDataCol As Long
62     Dim TextDataRow As Long, LastTextDataRow As Long
63     Dim FirstDataRow As Long, LastDataRow As Long, LastHeaderRow As Long
64     Dim LabelTemplate As Shape, TempLabelShape As Shape, LabelShape As Shape
65     Dim LabelRng As Range, LabelCell As Range
66     'Remove any labels currently on the sheet
67     For Each LabelShape In Sheet3.Shapes
68         LabelShape.Delete
69     Next
70     Set LabelShape = Nothing
71     'Remove any Label Preview (if existing)
72     On Error Resume Next
73     Sheet1.Shapes("LabelPreview").Delete
74     On Error GoTo 0
75
76     With Sheet1
77         TotalDataCol = .Range("J23").Value
78         FirstFiltCol = TotalDataCol + 5
79         LastFiltCol = FirstFiltCol + TotalDataCol
80         Range(Sheet4.Cells(1, FirstFiltCol), Sheet4.Cells(1, LastFiltCol)).Value = Range(Sheet4
81         .Cells(1, 1), Sheet4.Cells(1, TotalDataCol + 1)).Value 'Copy Over for headers criteria
82
83         Range(Sheet4.Cells(3, FirstFiltCol), Sheet4.Cells(3, LastFiltCol)).Value = Range(Sheet4
84         .Cells(1, 1), Sheet4.Cells(1, TotalDataCol + 1)).Value 'Copy Over headers for results
85         LastHeaderRow = Application.WorksheetFunction.CountA(.Range("D27:D77")) + 26 'Last
86         header Row
87         If LastHeaderRow < 27 Then Exit Sub 'On No Headers
88         .Range("G27:G" & LastHeaderRow).Copy
89         Sheet4.Cells(2, FirstFiltCol).PasteSpecial xlPasteValues, , , True 'Copy Over filters
90         LastDataRow = .Range("H22").Value - 1 'Last Row Of Data
91         Range(Sheet4.Cells(1, 1), Sheet4.Cells(LastDataRow, TotalDataCol + 1)).AdvancedFilter
92         xlFilterCopy, _
93         CriteriaRange:=Range(Sheet4.Cells(1, FirstFiltCol), Sheet4.Cells(2, LastFiltCol)),
94         CopyToRange:=Range(Sheet4.Cells(3, FirstFiltCol), Sheet4.Cells(3, LastFiltCol)), Unique
95         :=True
96
97         'Builld Labels On Label Page
98         On Error Resume Next
99         Set LabelTemplate = Sheet1.Shapes("TemplateGroup")
100        'LabelTemplate.Visible = msoCTrue
101        On Error GoTo 0
102        If LabelTemplate Is Nothing Then
103            GroupTemplateShapes 'Run Macro to Create Label Group
104            On Error Resume Next
105            Set LabelTemplate = Sheet1.Shapes("TemplateGroup") '2nd ry
106            On Error GoTo 0
107        End If
108        If LabelTemplate Is Nothing Then Exit Sub 'Exit on 2nd Failure to set locate label
109        template
110        FirstDataRow = 3
111        LastDataRow = Sheet4.Cells(99999, FirstFiltCol).End(xlUp).Row 'Last Results Row
112        .Range("E141").Value = LastDataRow - 3 'Display # Of labels to Print
113        .Range("G141").Value = Application.WorksheetFunction.RoundUp((LastDataRow - 3) / .
114        Range("E9").Value, 0) 'Total # Of Labels Divided by labels per sheet

```

1 2

```

1 2
106 TotalLabelCol = .Range("E11").Value 'Number of label columns
107 TotalLabelRow = Application.WorksheetFunction.RoundUp((LastDataRow - 3) / TotalLabelCol
, 0) ' .Range("G11").Value 'Number of label rows
108 LastLabelCol = TotalLabelCol + (TotalLabelCol - 1)
109 LastLabelRow = TotalLabelRow + (TotalLabelRow - 1)
110 LastTextDataRow = LastHeaderRow + 56
111 Sheet3.Activate
112 LabelRow = 1
113 LabelCol = 1
114 LabelNumber = 1
115 LabelTemplate.Copy
116 Sheet3.Activate
117
118 Sheet3.Range("A1").Select 'Paste in Temporary shape
119 Sheet3.Paste Destination:=Sheet3.Range("A1")
120 Set TempLabelShape = ThisWorkbook.Sheets("Printed Labels").Shapes("TemplateGroup")
121 TempLabelShape.Apply
122 Application.CutCopyMode = False
123
124 'First Loop Through Each Row Of Data
125 LabelDataRow = 4
126 For LabelRow = 1 To LastLabelRow Step 2 'First Loop Through Rows
127     For LabelCol = 1 To LastLabelCol Step 2 'Second Loop Through Columns
128         Set LabelCell = ThisWorkbook.Sheets("Printed Labels").Cells(LabelRow, LabelCol)
129
130         Sheet3.Activate
131         TempLabelShape.Copy
132         LabelCell.Select
133
134         On Error GoTo ErrorHandler
135         Sheet3.Paste Destination:=LabelCell
136
137         LabelDataCol = FirstFiltCol
138         For TextDataRow = 83 To LastTextDataRow 'Update Formulas For all Shapes if
selected
139             If .Range("G" & TextDataRow).Value = "ü" Then 'Show Data
140                 On Error Resume Next
141                 Sheet3.Shapes("Data" & TextDataRow).DrawingObject.Formula = "=" & Sheet4.Cells(LabelDataRow, LabelDataCol).Address & "Get Data"
Value
142                 Sheet3.Shapes("Data" & TextDataRow).Name = "Label" & LabelNumber &
"Data" & TextDataRow
143                 On Error GoTo 0
144             End If
145             LabelDataCol = LabelDataCol + 1 'Increment Data Column
146         Next TextDataRow
147
148         'If LabelShape Is Nothing Then GoTo PasteLabelAgain
149         With TempLabelShape
150             .Left = Sheet3.Cells(LabelRow, LabelCol).Left
151             .Top = Sheet3.Cells(LabelRow, LabelCol).Top
152             .Visible = msoCTrue
153             .Name = "Label" & LabelNumber
154         End With
155         Set TempLabelShape = Sheet3.Shapes("TemplateGroup") 'Reset Temp Label To
Original sheet Template
156
157
158         LabelNumber = LabelNumber + 1
159         LabelDataRow = LabelDataRow + 1
160     If LabelDataRow > LastDataRow Then Exit For

```

1 2 3 4

```
161     1 2 3 4
162         Next LabelCol
163         If LabelDataRow > LastDataRow Then Exit For
164     Next LabelRow
165     On Error Resume Next
166     Sheet3.Shapes("TemplateGroup" ).Delete 'Remove temporary group
167     On Error GoTo 0
168     If LabelRow > 20 Then LabelRow = 20 'Set max height for sample picture
169     Set LabelRng = Range(Sheet3.Cells(1, 1), Sheet3.Cells(LabelRow, LastLabelCol))
170     Application.CutCopyMode = False
171     LabelRng.CopyPicture
172     Sheet1.Activate
173     Sheet1.Range("D143" ).PasteSpecial
174     Selection.Formula = "'Printed Labels'!" & LabelRng.Address
175     Selection.Name = "LabelPreview"
176     Application.CutCopyMode = xlCopy
177     With .Shapes("LabelPreview" )
178         .IncrementLeft 20
179     End With
180     Sheet1.Range("F144" ).Select
181 End With
182 Exit Sub
183 ErrorHandler:
184 Application.Wait Now + 0.00001
185 TempLabelShape.Copy
186 Application.Wait Now + 0.00001
187 Sheet3.Paste Destination:=LabelCell
188 Resume Next
189 End Sub
190
191 Sub PrintLabels()
192     Sheet3.PrintOut , , , False, True, , , , False
193 End Sub
194
195
196
197
```

—
→, 16

A

Activate, 17, 18
Address, 15, 17, 18
AdvancedFilter, 16
Application, 15-18
Apply, 17

B

BottomMargin, 15
BuildLabelSheet, 15

C

Cells, 15-18
ColumnSpace, 15
ColumnWidth, 15
Copy, 16-18
CopyPicture, 18
CopyToRange, 16
CountA, 16
CriteriaRange, 16
CutCopyMode, 17, 18

D

DataRow, 15
Delete, 16, 18
Destination, 17, 18
DrawingObject, 17

E

ErrorHandler, 17, 18
Explicit, 15

F

FilterDataCreateLabels, 16
FirstDataRow, 16
FirstFiltCol, 16, 17
FirstLabelCol, 16
FirstLabelRow, 16
FooterMargin, 15
Formula, 17, 18

G

GroupTemplateShapes, 16

H

HeaderMargin, 15

I

InchesToPoints, 15
IncrementLeft, 18

L

LabelCell, 16-18
LabelCol, 15-18
LabelDataCol, 16, 17
LabelDataRow, 16-18
LabelNumber, 16, 17
LabelRng, 16, 18
LabelRow, 15-18
LabelShape, 16
LabelTemplate, 16, 17
LabHeight, 15
LabSheetQty, 15
LabWidth, 15
LastDataRow, 15-18
LastFiltCol, 16

LastHeaderRow, 16, 17
LastLabelCol, 15-18
LastLabelRow, 15-17
LastTextDataRow, 16, 17
Left, 17
LeftMargin, 15

M

MaxLabels, 15
MaxRows, 15
msoCTrue, 17

N

Name, 17, 18
Now, 18

P

PageSetup, 15
Paste, 17, 18
PasteSpecial, 16, 18
PrintArea, 15
PrintLabels, 18
PrintOut, 18

R

Range, 15-18
ResetCalc, 16
RightMargin, 15
RoundUp, 15-17
Row, 16
RowHeight, 15
RowSpace, 15

S

Selection, 18
SetCol, 15
SetCount, 15
SetRow, 15
Shape, 16
Shapes, 16-18
Sheet1, 15, 16, 18
Sheet3, 15-18
Sheet4, 16, 17
Sheets, 17
StopCalc, 15

T

TempLabelShape, 16-18
TextDataRow, 16, 17
ThisWorkbook, 17
Top, 17
TopMargin, 15
TotalDataCol, 16
TotalLabelCol, 16, 17
TotalLabelRow, 16, 17

U

Unique, 16

V

Value, 15-17
Visible, 17

W

Wait, 18
Width, 15
WorksheetFunction, 15-17

X

xlCopy, 18

xlFilterCopy, 16
xlPasteValues, 16
xlUp, 16

```

1  Option Explicit
2
3  Sub BrowseForData()
4      Dim FileFldr As FileDialog
5      Dim DataFilePath As String
6      Dim DataWs As Worksheet
7      Dim ThisWkBk As Workbook, DataWkBk As Workbook
8      Dim WsRow As Long
9
10     Set ThisWkBk = ThisWorkbook
11     With Sheet1
12         WsRow = 20
13         .Range("A20:A50,E20:H20,I21:J21,F22:F23,H22:H23").ClearContents 'Clear Any older data
14         .Range("D27:J77").ClearContents 'Clear any older Worksheet data
15         Set FileFldr = Application.FileDialog(msoFileDialogFilePicker)
16         With FileFldr
17             .Title = "Select Data File"
18             .Filters.Add "Excel Files", "*.xlsm,*.xlsx,*.xlsb,*.xls"
19             If .Show <> -1 Then GoTo NoSelection
20             Sheet1.Range("E20").Value = .SelectedItems(1) 'Get Full File Path
21         End With
22
23         DataFilePath = .Range("E20").Value 'Long File Path
24         .Range("E21").Value = Dir(DataFilePath, vbDirectory) 'File Name
25         Workbooks.Open (DataFilePath) 'Open Data Workbook
26         Set DataWkBk = Workbooks(Dir(DataFilePath)) 'Set Data Workbook
27
28         For Each DataWs In DataWkBk.Worksheets
29             Sheet1.Range("A" & WsRow).Value = DataWs.Name
30             WsRow = WsRow + 1
31         Next
32         ThisWkBk.Activate
33         .Range("I21").Select 'Select cell for user to select sheet
34         NoSelection:
35     End With
36 End Sub
37
38
39 Sub GetDataDetails()
40     Dim DataFilePath As String, DataShtName As String
41     Dim WsRow As Long, FirstRow As Long, TableRow As Long, TableCol As Long, FirstCol As Long,
42     LastRow As Long, LastCol As Long
43     Dim DataRow As Long, DataCol As Long, DesignRow As Long
44     Dim DataWkBk As Workbook, ThisWkBk As Workbook
45
46     Set ThisWkBk = ThisWorkbook
47     With Sheet1
48         DataFilePath = .Range("E20").Value 'Data Workbook File Path
49         On Error GoTo MissingWorkbook
50         Set DataWkBk = Workbooks(Dir(DataFilePath)) 'Set Data Workbook
51         On Error GoTo 0
52         DataShtName = .Range("I21").Value 'Data Worksheet Name
53         .Range("D27:J77").ClearContents 'Clear any older Worksheet data
54         .Range("D83:G132").ClearContents 'Clear any Lables & Data for design steps
55     End With
56
57     With DataWkBk.Sheets(DataShtName)
58
59         'Get Last Row Of Data

```

```

1 2
60 'On Error Resume Next
61 LastRow = .Cells.Find(What:="*", After:=.Range("A1"), _
62 Lookat:=xlPart, LookIn:=xlFormulas, SearchOrder:=xlByRows, SearchDirection:=xlPrevious,
    MatchCase:=False).Row
63 'Get Last Column Of Data
64 LastCol = .Cells.Find(What:="*", After:=.Range("A1"), _
65 Lookat:=xlPart, LookIn:=xlFormulas, SearchOrder:=xlByColumns, SearchDirection:=
    xlPrevious, MatchCase:=False).Column
66 On Error GoTo 0
67
68 'Determine First Row Of Table w/ 70% of the cells filled
69 For TableRow = 1 To LastRow
70     If (Application.WorksheetFunction.CountA(Range(.Cells(TableRow, 1), .Cells(TableRow,
71         LastCol))) / LastCol) >= 0.7 Then
72         FirstRow = TableRow
73         Exit For
74     End If
75 Next TableRow
76
77 'Determine First Column Of Table w/ 50% of the cells filled
78 For TableCol = 1 To LastCol
79     If (Application.WorksheetFunction.CountA(Range(.Cells(1, TableCol), .Cells(LastRow,
80         TableCol))) / LastRow) >= 0.5 Then
81         FirstCol = TableCol
82         Exit For
83     End If
84 Next TableCol
85 End With
86
87 If Sheet1.Range("B18").Value = True Then FirstRow = FirstRow + 1 'Has Headers
88
89 Sheet1.Range("F22").Value = FirstRow 'Set First Row Of Data Data
90 Sheet1.Range("H22").Value = LastRow 'Set Last Row Of Data Data
91 Sheet1.Range("F23").Value = FirstCol 'Set First Column Of Data Data
92 Sheet1.Range("H23").Value = LastCol 'Set Last Column Of Data Data
93
94 ThisWkBk.Activate
95 'Load Data Details
96 DataRow = 27
97 DesignRow = 83
98 With Sheet1
99     If .Range("B18").Value = True Then 'Data has Headers
100
101         For DataCol = FirstCol To LastCol
102             .Range("D" & DataRow).Value = DataWkBk.Sheets(DataShtName).Cells(FirstRow - 1,
103                 DataCol).Value 'Load Headers
104             .Range("D" & DesignRow).Value = DataWkBk.Sheets(DataShtName).Cells(FirstRow - 1,
105                 DataCol).Value 'Load Headers
106             DataRow = DataRow + 1
107             DesignRow = DesignRow + 1
108         Next DataCol
109     End If
110     'Add First Row Of Data
111     DataRow = 27
112     DesignRow = 83
113     For DataCol = FirstCol To LastCol
114         .Range("E" & DataRow).Value = DataWkBk.Sheets(DataShtName).Cells(FirstRow, DataCol)
115         .Value
116         .Range("E" & DesignRow).Value = DataWkBk.Sheets(DataShtName).Cells(FirstRow,
117             DataCol).Value 'Load First Row OF Data

```

1 2 3


```
113     1 2 3     DataRow = DataRow + 1
114           DesignRow = DesignRow + 1
115           Next DataCol
116     End With
117     Exit Sub
118     MissingWorkbook:
119     MsgBox "Please check for a correct Data Workbook"
120     End Sub
121
122
123
```

—
→, 21

A

Activate, 20, 21
Add, 20
After, 21
Application, 20, 21

B

BrowseForData, 20

C

Cells, 21
ClearContents, 20
Column, 21
CountA, 21

D

DataCol, 20-22
DataFilePath, 20
DataRow, 20-22
DataShtName, 20, 21
DataWkBk, 20, 21
DataWs, 20
DesignRow, 20-22
Dir, 20

E

Explicit, 20

F

FileDialog, 20
FileFldr, 20
Filters, 20
Find, 21
FirstCol, 20, 21
FirstRow, 20, 21

G

GetDataDetails, 20

L

LastCol, 20, 21
LastRow, 20, 21
Lookat, 21
LookIn, 21

M

MatchCase, 21
MissingWorkbook, 20, 22
MsgBox, 22
msoFileDialogFilePicker, 20

N

Name, 20
NoSelection, 20

O

Open, 20

R

Range, 20, 21
Row, 21

S

SearchDirection, 21
SearchOrder, 21
SelectedItems, 20

Sheet1, 20, 21
Sheets, 20, 21
Show, 20

T

TableCol, 20, 21
TableRow, 20, 21
ThisWkBk, 20, 21
ThisWorkbook, 20
Title, 20

V

Value, 20, 21
vbDirectory, 20

W

What, 21
Workbook, 20
Workbooks, 20
Worksheet, 20
WorksheetFunction, 21
Worksheets, 20
WsRow, 20

X

xlByColumns, 21
xlByRows, 21
xlFormulas, 21
xlPart, 21
xlPrevious, 21

```
1 Option Explicit
2
3
4 Sub AddFilt_Numb_GreaterThan()
5     Dim DataRow As Long
6     Dim FilterBy As String
7     With Sheet1
8         If .Range("B1").Value = Empty Then Exit Sub
9         DataRow = .Range("B1").Value
10        If .Range("G" & DataRow).Value = Empty Then Exit Sub
11        FilterBy = .Range("G" & DataRow).Value 'Filter By Text
12        FilterBy = Replace(Replace(Replace(FilterBy, "*", "" ), "<" , "" ), ">" , "" )
13        'Remove any previous filtering characters
14        .Range("G" & DataRow).Value = ">" & FilterBy
15    End With
16 End Sub
17
18 Sub AddFilt_Numb_LessThan()
19     Dim DataRow As Long
20     Dim FilterBy As String
21     With Sheet1
22         If .Range("B1").Value = Empty Then Exit Sub
23         DataRow = .Range("B1").Value
24         If .Range("G" & DataRow).Value = Empty Then Exit Sub
25         FilterBy = .Range("G" & DataRow).Value 'Filter By Text
26         FilterBy = Replace(Replace(Replace(FilterBy, "*", "" ), "<" , "" ), ">" , "" )
27         'Remove any previous filtering characters
28         .Range("G" & DataRow).Value = "<" & FilterBy
29    End With
30 End Sub
31
32 Sub AddFilt_Text_Contains()
33     Dim DataRow As Long
34     Dim FilterBy As String
35     With Sheet1
36         If .Range("B1").Value = Empty Then Exit Sub
37         DataRow = .Range("B1").Value
38         If .Range("G" & DataRow).Value = Empty Then Exit Sub
39         FilterBy = .Range("G" & DataRow).Value 'Filter By Text
40         FilterBy = Replace(Replace(Replace(FilterBy, "*", "" ), "<" , "" ), ">" , "" )
41         'Remove any previous filtering characters
42         .Range("G" & DataRow).Value = "*" & FilterBy & "*"
43    End With
44 End Sub
45
46 Sub AddFilt_Text_EndsWith()
47     Dim DataRow As Long
48     Dim FilterBy As String
49     With Sheet1
50         If .Range("B1").Value = Empty Then Exit Sub
51         DataRow = .Range("B1").Value
52         If .Range("G" & DataRow).Value = Empty Then Exit Sub
53         FilterBy = .Range("G" & DataRow).Value 'Filter By Text
54         FilterBy = Replace(Replace(Replace(FilterBy, "*", "" ), "<" , "" ), ">" , "" )
55         'Remove any previous filtering characters
56         .Range("G" & DataRow).Value = "*" & FilterBy
57    End With
58 End Sub
59
60 Sub AddFilt_Text_StartsWith()
61     Dim DataRow As Long
62     Dim FilterBy As String
```

```
1
58 With Sheet1
59   If .Range("B1").Value = Empty Then Exit Sub
60   DataRow = .Range("B1").Value
61   If .Range("G" & DataRow).Value = Empty Then Exit Sub
62   FilterBy = .Range("G" & DataRow).Value 'Filter By Text
63   FilterBy = Replace(Replace(Replace(FilterBy, "*", "" ), "<" , "" ), ">" , "" )
        'Remove any previous filtering characters
64   .Range("G" & DataRow).Value = FilterBy & "*"
65 End With
66 End Sub
67
68 Sub ShowFilterGroup()
69   Dim DataRow As Long
70   With Sheet1
71     If .Range("B1").Value = Empty Then Exit Sub
72     DataRow = .Range("B1").Value 'Data Row
73     With .Shapes("FiltGrp" )
74       .Left = Sheet1.Range("H" & DataRow).Left
75       .Top = Sheet1.Range("H" & DataRow).Top
76       .IncrementTop -15
77       .Visible = msoCTrue
78     End With
79   End With
80 End Sub
```

A

AddFilt_Numb_GreaterThan, [24](#)
AddFilt_Numb_LessThan, [24](#)
AddFilt_Text_Contains, [24](#)
AddFilt_Text_EndsWith, [24](#)
AddFilt_Text_StartsWith, [24](#)

D

DataRow, [24](#), [25](#)

E

Empty, [24](#), [25](#)
Explicit, [24](#)

F

FilterBy, [24](#), [25](#)

I

IncrementTop, [25](#)

L

Left, [25](#)

M

msoCTrue, [25](#)

R

Range, [24](#), [25](#)
Replace, [24](#), [25](#)

S

Shapes, [25](#)
Sheet1, [24](#), [25](#)
ShowFilterGroup, [25](#)

T

Top, [25](#)

V

Value, [24](#), [25](#)
Visible, [25](#)

```
1 Option Explicit
2
3
4 Sub LoadLabel()
5     With Sheet1
6         Dim LabelRow As Long
7         Dim PicName As String
8         If .Range("B5").Value = Empty Then
9             MsgBox "Please select a correct label from the drop down list"
10            Exit Sub
11        End If
12        LabelRow = .Range("B5").Value 'Label Row
13        .Range("E9").Value = Sheet2.Range("B" & LabelRow).Value '# Labels Per Sheet
14        .Range("G9").Value = Sheet2.Range("C" & LabelRow).Value 'Type
15        .Range("E10").Value = Sheet2.Range("D" & LabelRow).Value 'Height
16        .Range("G10").Value = Sheet2.Range("E" & LabelRow).Value 'Width
17        .Range("E11").Value = Sheet2.Range("F" & LabelRow).Value 'Columns
18        .Range("G11").Value = Sheet2.Range("G" & LabelRow).Value 'Rows
19        .Range("E13").Value = Sheet2.Range("I" & LabelRow).Value 'Column Spacing
20        .Range("G13").Value = Sheet2.Range("J" & LabelRow).Value 'Row Spacing
21        .Range("E14").Value = Sheet2.Range("K" & LabelRow).Value 'Left Margin
22        .Range("G14").Value = Sheet2.Range("L" & LabelRow).Value 'Right Margin
23        .Range("E15").Value = Sheet2.Range("M" & LabelRow).Value 'Top Margin
24        .Range("G15").Value = Sheet2.Range("N" & LabelRow).Value 'Bottom Margin
25        PicName = Sheet2.Range("H" & LabelRow).Value 'Filename
26        Sheet2.Shapes(PicName).Copy
27        .Range("H8").Select
28        .Paste
29        On Error Resume Next
30        .Shapes("LabSheetPic").Delete
31        On Error GoTo 0
32        With .Shapes(PicName)
33            .Name = "LabSheetPic"
34            .Left = Sheet1.Range("H8").Left
35            .Top = Sheet1.Range("H8").Top
36            .LockAspectRatio = msoCTrue
37            .Width = 90
38            .IncrementLeft 40
39        End With
40        .Range("E8").Select
41    End With
End Sub
```

C

Copy, [27](#)

D

Delete, [27](#)

E

Empty, [27](#)

Explicit, [27](#)

I

IncrementLeft, [27](#)

L

LabelRow, [27](#)

Left, [27](#)

LoadLabel, [27](#)

LockAspectRatio, [27](#)

M

MsgBox, [27](#)

msoCTrue, [27](#)

N

Name, [27](#)

P

Paste, [27](#)

PicName, [27](#)

R

Range, [27](#)

S

Shapes, [27](#)

Sheet1, [27](#)

Sheet2, [27](#)

T

Top, [27](#)

V

Value, [27](#)

W

Width, [27](#)

```

1  Option Explicit
2
3  Sub ImportLabelData()
4      Dim DataFilePath As String, DataShtName As String
5      Dim DataWkBk As Workbook, ThisWkBk As Workbook
6
7      Dim FirstDataRow As Long, FirstDataCol As Long, LastDataCol As Long
8      Dim LastDataRow As Long, DataCol As Long, DataRow As Long
9
10     Dim TotalDataRows As Long, TotalDataColumns As Long
11     Dim TableRow As Long
12     Dim TableCol As Long
13     Dim ColCount As Long
14     Dim FirstAvailRow As Long
15
16
17     Dim MapCol As Long
18     Dim LastMapCol As Long
19     Dim CellFormat As String
20
21     With Sheet1
22         'Make sure an proper file path
23         DataFilePath = .Range("E20").Value 'Date File Path
24         If Dir(DataFilePath) = "" Then
25             MsgBox "Please select a proper data file for your labels"
26             CreateLabelStep2
27             Exit Sub
28         End If
29
30         Set ThisWkBk = ThisWorkbook 'Set Current Workbook To Import
31
32         'Get Data Workbook Variables
33         On Error Resume Next 'An error will occur if we set a workbook that is not open. If
34         the error occurs we can run a check and open the workbook
35         Set DataWkBk = Workbooks(Dir(DataFilePath)) 'Set Data Workbook
36         If DataWkBk Is Nothing Then 'Data Workbook is not open
37             Set DataWkBk = Workbooks.Open(DataFilePath)
38         End If
39         DataShtName = .Range("I21").Value 'Data Worksheet Name
40         FirstDataRow = .Range("F22").Value 'First Row of Data (excludes headers)
41         FirstDataCol = .Range("F23").Value 'First Column Of Data from the Data sheet
42         LastDataCol = .Range("H23").Value 'Last Data Column
43         LastDataRow = .Range("H22").Value 'Last Row Of Data
44         TotalDataRows = .Range("J22").Value 'Total Rows To Be copied
45         TotalDataColumns = .Range("J23").Value 'Total Columns To be copied
46         Sheet4.Range("A1:ZZ9999").ClearContents 'Clear Any Existing Data
47
48         'Copy Over headers (if any)
49         If .Range("B18").Value = True Then Range(Sheet4.Cells(1, 1), Sheet4.Cells(1,
50         TotalDataColumns)).Value = _
51         Range(DataWkBk.Sheets(DataShtName).Cells(FirstDataRow - 1, FirstDataCol), DataWkBk.
52         Sheets(DataShtName).Cells(FirstDataRow - 1, LastDataCol)).Value
53
54         'Copy Over Data
55         Range(Sheet4.Cells(2, 1), Sheet4.Cells(TotalDataRows + 1, TotalDataColumns)).Value = _
56         Range(DataWkBk.Sheets(DataShtName).Cells(FirstDataRow, FirstDataCol), DataWkBk.Sheets(
57         DataShtName).Cells(LastDataRow, LastDataCol)).Value
58
59     End With
60
61     Exit Sub
62     DuplicateSheet:
63     MsgBox "The Sheet Name '" & .Range("I30").Value & "' already exists" & vbCrLf &

```



```
58     1 2 "Please select a new worksheet name to add"  
59     End With  
60     End Sub  
61  
62  
63
```

—
→, 29

C

CellFormat, 29
Cells, 29
ClearContents, 29
ColCount, 29
CreateLabelStep2, 29

D

DataCol, 29
DataFilePath, 29
DataRow, 29
DataShtName, 29
DataWkBk, 29
Dir, 29
DuplicateSheet, 29

E

Explicit, 29

F

FirstAvailRow, 29
FirstDataCol, 29
FirstDataRow, 29

I

ImportLabelData, 29

L

LastDataCol, 29
LastDataRow, 29
LastMapCol, 29

M

MapCol, 29
MsgBox, 29

O

Open, 29

R

Range, 29

S

Sheet1, 29
Sheet4, 29
Sheets, 29

T

TableCol, 29
TableRow, 29
ThisWkBk, 29
ThisWorkbook, 29
TotalDataColumns, 29
TotalDataRows, 29

V

Value, 29
vbCrLf, 29

W

Workbook, 29
Workbooks, 29

```
1 Option Explicit
2
3 Sub GroupTemplateShapes()
4     With Sheet1
5         Dim LabelShape As Shape, TemplateGroup As Shape
6         Dim ShapesNames As String
7         Dim GrpArr As Variant
8         For Each LabelShape In Sheet1.Shapes
9             If InStr(LabelShape.Name, "Data" ) > 0 Or InStr(LabelShape.Name, "Label" ) > 0 Then
10                ShapesNames = ShapesNames & LabelShape.Name & ","
11            End If
12        Next LabelShape
13        If ShapesNames = Empty Then Exit Sub 'Grouped already or no shapes exist
14        ShapesNames = Left(ShapesNames, Len(ShapesNames) - 1)
15        GrpArr = Split(ShapesNames, "," )
16        On Error GoTo NoGroup
17        Set TemplateGroup = ThisWorkbook.Sheets(1).Shapes.Range((GrpArr)).Group
18        With TemplateGroup
19            .Name = "TemplateGroup"
20            .LockAspectRatio = msoCTrue
21            .Placement = xlFreeFloating
22            .Visible = msoFalse
23        End With
24    End With
25    NoGroup:
26 End Sub
27
28
29 Sub UngroupTemplateShape()
30     On Error Resume Next
31     With Sheet1.Shapes("TemplateGroup" )
32         .Visible = msoCTrue
33         .Ungroup
34     End With
35     On Error GoTo 0
36 End Sub
```

E

Empty, [32](#)
Explicit, [32](#)

G

Group, [32](#)
GroupTemplateShapes, [32](#)
GrpArr, [32](#)

I

InStr, [32](#)

L

LabelShape, [32](#)
Left, [32](#)
Len, [32](#)
LockAspectRatio, [32](#)

M

msoCTrue, [32](#)
msoFalse, [32](#)

N

Name, [32](#)
NoGroup, [32](#)

P

Placement, [32](#)

R

Range, [32](#)

S

Shape, [32](#)
Shapes, [32](#)
ShapesNames, [32](#)
Sheet1, [32](#)
Sheets, [32](#)
Split, [32](#)

T

TemplateGroup, [32](#)
ThisWorkbook, [32](#)

U

Ungroup, [32](#)
UngroupTemplateShape, [32](#)

V

Visible, [32](#)

X

xlFreeFloating, [32](#)

```
1 Option Explicit
2
3 Sub CreateLabelStep1()
4 StopCalc 'Import Step 1
5 GroupTemplateShapes
6 ResetHideSteps
7 Sheet1.Shapes("Step1").Fill.ForeColor.RGB = RGB(50, 255, 50)
8 Sheet1.Shapes("Step1").TextFrame2.TextRange.Font.Fill.ForeColor.RGB = RGB(0, 0, 0)
9 Sheet1.Range("5:16").EntireRow.Hidden = False
10 Sheet1.Shapes("Step1Grp").Visible = msoCTrue
11 On Error Resume Next
12 Sheet1.Shapes("LabSheetPic").Visible = msoCTrue
13 Sheet1.Range("E8").Select
14 On Error GoTo 0
15 ResetCalc
16 End Sub
17
18 Sub CreateLabelStep2()
19 StopCalc 'Import Step 2
20 GroupTemplateShapes
21 ResetHideSteps
22 Sheet1.Range("17:77").EntireRow.Hidden = False
23 Sheet1.Shapes("Step2").Fill.ForeColor.RGB = RGB(50, 255, 50)
24 Sheet1.Shapes("Step2").TextFrame2.TextRange.Font.Fill.ForeColor.RGB = RGB(0, 0, 0)
25 Sheet1.Shapes("Step2Grp").Visible = msoCTrue
26 Sheet1.Range("E20").Select
27 ResetCalc
28 End Sub
29
30 Sub CreateLabelStep3()
31 If Sheet1.Range("E20").Value = Empty Or Sheet1.Range("E21").Value = Empty Then
32 MsgBox "Please select a data file and sheet for your labels before moving onto the
33 next step"
34 Exit Sub
35 End If
36 StopCalc 'Import Step 3
37 ResetHideSteps
38 Sheet1.Range("78:132").EntireRow.Hidden = False
39 Sheet1.Shapes("Step3").Fill.ForeColor.RGB = RGB(50, 255, 50)
40 Sheet1.Shapes("Step3").TextFrame2.TextRange.Font.Fill.ForeColor.RGB = RGB(0, 0, 0)
41 Sheet1.Shapes("Step3Grp").Visible = msoCTrue
42 UngroupTemplateShape 'Ungroup any lable shapes (if existing)
43 With Sheet1.Shapes("BackLabShape")
44 .Height = Application.InchesToPoints(Sheet1.Range("E10").Value)
45 .Width = Application.InchesToPoints(Sheet1.Range("G10").Value)
46 .Visible = msoCTrue
47 End With
48 ImportLabelData
49 ResetCalc
50 End Sub
51
52 Sub CreateLabelStep4()
53 If Sheet1.Range("E20").Value = Empty Or Sheet1.Range("E21").Value = Empty Then
54 MsgBox "Please select a data file and sheet for your labels before moving onto the
55 next step"
56 Exit Sub
57 End If
58 StopCalc 'Import Step 4
59 GroupTemplateShapes 'Group Shapes before moving to Step 4
60 ResetHideSteps
61 Sheet1.Range("133:180").EntireRow.Hidden = False
```

1

```
1
60 Sheet1.Shapes("Step4" ).Fill.ForeColor.RGB = RGB(50, 255, 50)
61 Sheet1.Shapes("Step4" ).TextFrame2.TextRange.Font.Fill.ForeColor.RGB = RGB(0, 0, 0)
62 Sheet1.Shapes("Step4Grp" ).Visible = msoCTrue
63 BuildLabelSheet
64 ResetCalc
65 FilterDataCreateLabels
66 End Sub
67
68 Sub ResetHideSteps()
69 With Sheet1 'Hide All Shapes & Rows
70 .Range("5:180" ).EntireRow.Hidden = True
71 .Shapes.Range(Array("ButtonSet" )).Fill.ForeColor.RGB = RGB(0, 97, 0)
72 .Shapes.Range(Array("ButtonSet" )).TextFrame2.TextRange.Font.Fill.ForeColor.RGB = RGB(
255, 255, 255)
73 .Shapes("Step1Grp" ).Visible = msoFalse
74 .Shapes("Step2Grp" ).Visible = msoFalse
75 .Shapes("Step3Grp" ).Visible = msoFalse
76 .Shapes("Step4Grp" ).Visible = msoFalse
77 On Error Resume Next
78 Sheet1.Shapes("LabSheetPic" ).Visible = msoFalse 'Hide Label Sheet Icon
79 Sheet1.Shapes("BackLabShape" ).Visible = msoFalse 'Hide LabelTemplate Background
80 Sheet1.Shapes("LabelPreview" ).Visible = msoFalse
81 On Error GoTo 0
82 End With
83 End Sub
```

A

Application, 34
Array, 35

B

BuildLabelSheet, 35

C

CreateLabelStep1, 34
CreateLabelStep2, 34
CreateLabelStep3, 34
CreateLabelStep4, 34

E

Empty, 34
EntireRow, 34, 35
Explicit, 34

F

Fill, 34, 35
FilterDataCreateLabels, 35
Font, 34, 35
ForeColor, 34, 35

G

GroupTemplateShapes, 34

H

Height, 34
Hidden, 34, 35

I

ImportLabelData, 34
InchesToPoints, 34

M

MsgBox, 34
msoCTrue, 34, 35
msoFalse, 35

R

Range, 34, 35
ResetCalc, 34, 35
ResetHideSteps, 34, 35
RGB, 34, 35

S

Shapes, 34, 35
Sheet1, 34, 35
StopCalc, 34

T

TextFrame2, 34, 35
TextRange, 34, 35

U

UngroupTemplateShape, 34

V

Value, 34
Visible, 34, 35

W

Width, 34

Thank You!

This source code was created and made available to help you gain a better understanding of how VBA is used to create amazing Excel-based applications.

Thank you so much for your continued shares, likes and support. It really helps.



Excel For Freelancers