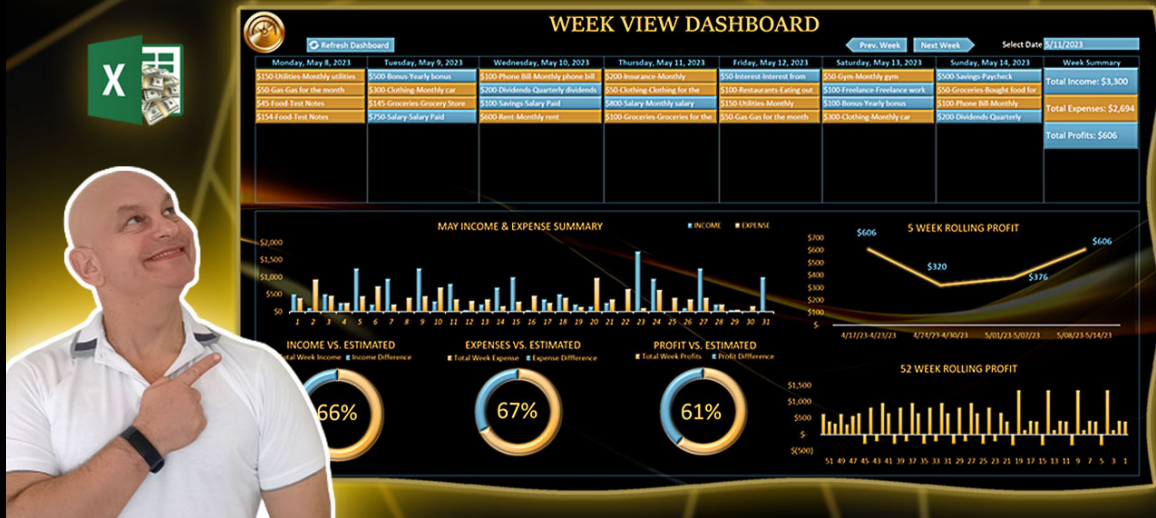


VBA SOURCE CODE BOOK

WEEK VIEW DASHBOARD



 *Excel For Freelancers*

Create This Powerful Week
View Dashboard From
Scratch



[DOWNLOAD
APPLICATION](#)



[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 over 442,000 YouTube subscribers, 33,000,000 video views, 250+ comprehensive training videos, and a thriving 60,000 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](#)



[DISCORD](#)



[INSTAGRAM](#)



[TELEGRAM](#)



[RUMBLE](#)



Microsoft®
Most Valuable
Professional



OUR COURSES & PRODUCTS



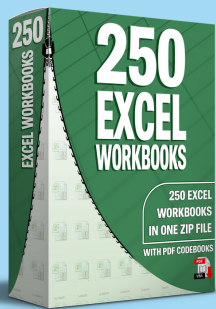
This comprehensive program will take you through a 12-phase process that will turn your enthusiasm for Excel into passive income.

[Click here to learn more](#)



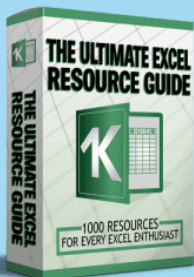
16 hour masterclass that will teach you the tips, tricks and techniques on how to create a dynamic single-click dashboard, and a ton more

[Click here to learn more](#)



Incredible Package of 250 of my BEST Applications now with PDF VBA Codebooks packed into a SINGLE ZIP File which also includes the "250 Workbook Library".

[Click here to learn more](#)



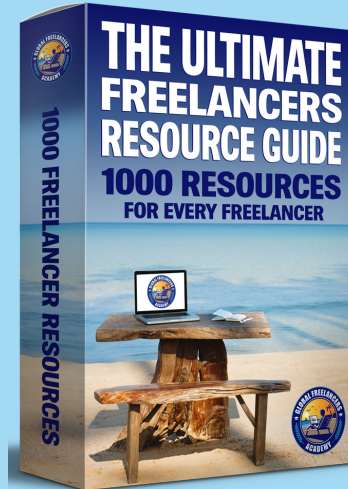
With 1000 live links, continuously updating content, sort-able and filterable items, you will always have exactly what you need, when you need it.

[Click here to learn more](#)



This incredible 13-hour freelancer masterclass will teach anyone how to be a successful freelancer with my proven 9-Phase 'Financial Freedom Roadmap' and includes 30+ downloads and exercises along with a ton of resources for anyone to reach their goals through freelancing.

[Click here to learn more](#)



I've put together all the Freelancing essentials, from freelance tools to freelance templates, in this Ultimate Freelancer's Resource Guide for freelancers at any stage in their career. With 1,000 Live Links, and a single click-to-update application, you will always have the most current and up-to-date information at your fingertips.

[Click here to learn more](#)

Table of Contents

Projects.....	2
VBAProject.....	2
Documents.....	2
Admin.....	2
(Declarations).....	2
Dashboard.....	3
(Declarations).....	3
Worksheet_Change [Sub].....	3
Worksheet_SelectionChange [Sub].....	3
Sheet4.....	4
(Declarations).....	4
ThisWorkbook.....	5
(Declarations).....	5
Transactions.....	6
(Declarations).....	6
Modules.....	7
Dashboard_Macros.....	7
(Declarations).....	7
Dashboard_Refresh [Sub].....	7
Next_Week [Sub].....	8
Prev_Week [Sub].....	8
PopUp_Calendar.....	9
(Declarations).....	9
CalCol [Sub].....	9
CalendarHide [Sub].....	9
CalendarShow [Sub].....	9
CalFormulaReplacement [Sub].....	10
CheckForSheet [Sub].....	10
CreateCalSht [Sub].....	11
GroupCal [Sub].....	13
MacroLinkRemover [Sub].....	13
NextMonth [Sub].....	14
NextYear [Sub].....	14
PrevMonth [Sub].....	14
PrevYear [Sub].....	15
ReplaceCalendar [Sub].....	15
SelectDay [Sub].....	15
ShowSettings [Sub].....	15
UnGroupCal [Sub].....	15

1	Option Explicit
2	

```
1 Option Explicit
2
3 Private Sub Worksheet_Change(ByVal Target As Range)
4     If Not Intersect(Target, Range("K2" )) Is Nothing And Range("K2" ).Value <> Empty
5         Then
6             Range("B4" ).Value = Range("K2" ).Value 'Entered Date
7             Dashboard_Refresh
8         End If
9     End Sub
10
11 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
12     Dim Cal As Shape
13     On Error Resume Next
14     Set Cal = Shapes("Calendar" )
15     On Error GoTo 0
16     If Cal Is Nothing Then ReplaceCalendar 'Shape Delete Copy from Popup Sheet
17     If Shapes("Calendar" ).Visible = True Then Shapes("Calendar" ).Visible = msoFalse
18     If Target.CountLarge > 1 Then Exit Sub 'If user selects more than 1 cell, then exit
19     sub
20         If Not Intersect(Target, Range("K2" )) Is Nothing Then
21             CheckForSheet
22             CalendarShow 'Macro to display Calenar
23         Else
24             CheckForSheet
25             CalendarHide
26         End If
27     End Sub
28 End Sub
```

1
2

Option Explicit

1 Option Explicit

2

1
2

Option Explicit

```

1 Option Explicit
2 Dim LastRow As Long, LastResultRow As Long, ResultRow As Long, WeekCol As Long, WeekRow
  As Long
3 Dim TransType As String, TransAmt As String, TransAcct As String, TransID As String,
  TransNotes As String
4 Dim TransShp As Shape
5
6 Sub Dashboard_Refresh()
7     With Dashboard
8         'Clear each Transaction Shape in Dashboard
9         For Each TransShp In Dashboard.Shapes
10            If InStr(TransShp.Name, "TransItem" ) > 0 Then TransShp.Delete 'Delete any
              transaction shape
11        Next TransShp
12    End With
13    With Transactions
14        LastRow = .Range("A99999" ).End(xlUp).Row 'Last Row Of data
15        If LastRow < 3 Then Exit Sub
16        Application.ScreenUpdating = False
17        Application.Calculation = xlCalculationManual
18        For WeekCol = 4 To 10
19            WeekRow = 5 'Set Intial Week Row
20            .Range("H3" ).Value = Dashboard.Cells(4, WeekCol).Value 'Day Date Criteria
21            .Range("A2:F" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
              "H2:H3" ), CopyToRange:=.Range("J2:N2" ), Unique:=True
22            LastResultRow = .Range("J99999" ).End(xlUp).Row
23            If LastResultRow < 3 Then GoTo NextDay
24            For ResultRow = 3 To LastResultRow
25                TransID = .Range("J" & ResultRow).Value 'Trans. ID
26                TransType = .Range("K" & ResultRow).Value 'Type
27                TransAcct = .Range("L" & ResultRow).Value 'Trans ACCOUNT
28                TransAmt = Format(.Range("M" & ResultRow).Value, "$#,###" ) 'Amount
29                TransNotes = .Range("N" & ResultRow).Value 'Notes
30                If TransType = "Income" Then 'Income Type
31                    Dashboard.Shapes("IncSampleShp" ).Duplicate.Name = "TransItem" &
                      TransID
32                Else 'Expenses Type
33                    Dashboard.Shapes("ExpSampleShp" ).Duplicate.Name = "TransItem" &
                      TransID
34                End If
35                With Dashboard.Shapes("TransItem" & TransID)
36                    .Left = Dashboard.Cells(WeekRow, WeekCol).Left
37                    .Top = Dashboard.Cells(WeekRow, WeekCol).Top + 2
38                    .Height = Dashboard.Cells(WeekRow, WeekCol).Height 'Set to same height
39                    of row
40                    .Width = Dashboard.Cells(WeekRow, WeekCol).Width - 2
41                    .TextFrame2.TextRange.Text = TransAmt & "-" & TransAcct & "-" &
                      TransNotes 'Shape Text
42                    WeekRow = WeekRow + 1 'Increment Row
43                End With
44            Next ResultRow
45        Next WeekCol
46    End With
47    With Dashboard
48        With .Shapes("TotalIncShp" )
49            .Left = Dashboard.Range("K5" ).Left + 1
50            .Top = Dashboard.Range("K5" ).Top
51            .Width = Dashboard.Range("K:K" ).Width - 2
52            .Height = Dashboard.Range("K5:K6" ).Height
53            .TextFrame2.TextRange.Text = "Total Income: " & Format(Transactions.Range(

```

1 2 3

```
1 2 3
54     "Q4" ).Value, "$#,##" ) 'Set Total Income
55     End With
56     With .Shapes("TotalExpShp" )
57         .Left = Dashboard.Range("K5" ).Left + 1
58         .Top = Dashboard.Range("K7" ).Top
59         .Width = Dashboard.Range("K:K" ).Width - 2
60         .Height = Dashboard.Range("K7:K8" ).Height
61         .TextFrame2.TextRange.Text = "Total Expenses: " & Format(Transactions.Range(
62         "Q10" ).Value, "$#,##" ) 'Set Total Expenses
63     End With
64     With .Shapes("TotalProfitShp" )
65         .Left = Dashboard.Range("K5" ).Left + 1
66         .Top = Dashboard.Range("K9" ).Top
67         .Width = Dashboard.Range("K:K" ).Width - 2
68         .Height = Dashboard.Range("K9:K10" ).Height
69         .TextFrame2.TextRange.Text = "Total Profits: " & Format(Transactions.Range(
70         "Q16" ).Value, "$#,##" ) 'Set Total Profit
71     End With
72 End With
73 Application.Calculation = xlCalculationAutomatic
74 Application.ScreenUpdating = True
75 End Sub
76
77 Sub Next_Week()
78     Dashboard.Range("B4" ).Value = Dashboard.Range("B4" ).Value + 7
79     Dashboard_Refresh
80 End Sub
81
82 Sub Prev_Week()
83     Dashboard.Range("B4" ).Value = Dashboard.Range("B4" ).Value - 7
84     Dashboard_Refresh
85 End Sub
```

```

1 Dim SelCell As Range
2 Dim DayName As String
3
4
5
6 .....''Color Calendar Background''.....
7 Sub CalCol()
8     With ActiveSheet.Shapes.Range(Array("CalBack" , "Settings" )).Select
9         With Selection.ShapeRange.Fill
10            .ForeColor.RGB = ActiveSheet.Shapes(Application.Caller).Fill.ForeColor.RGB
11        End With
12        ActiveSheet.Range(Sheets("CalPopUp" ).Range("A7" ).Value).Select
13    End With
14 End Sub
15 Sub CalendarHide()
16     Dim DayNum As Long
17     'Hide Calendar, Reset Day Colors
18     On Error GoTo NoCal
19     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
20     Sheets("CalPopUp" ).Range("A7" ).Value = ""
21     If Sheets("CalPopUp" ).Range("A20" ).Value <> Empty Then
22         For DayNum = 1 To 42
23             DayName = DayNum & "Day"
24             With ActiveSheet.Shapes(DayName)
25                 .Fill.ForeColor.RGB = RGB(255, 255, 255)
26                 .TextFrame2.TextRange.Font.Bold = msoFalse
27             End With
28         Next DayNum
29     End If
30     Exit Sub
31     NoCal: 'If calendar has been removed by accident, paste in backup calendar from
           CalPopUp Sheet
32 End Sub
33
34 Sub CalendarShow()
35     With ActiveSheet
36         Set SelCell = ActiveCell
37         'Check if active cell is a valid date
38         If IsDate(SelCell.Value) = True Then
39             Sheets("CalPopUp" ).Range("A1" ).Value = SelCell.Value
40         Else: 'If No Date or incorrect Date user current date
41             Sheets("CalPopUp" ).Range("A1" ).Value = "=Today()"
42         End If
43         'Clear all shapes to white (if calendar is visible)
44         If ActiveSheet.Shapes("Calendar" ).Visible = True Then
45             For DayNum = 1 To 42
46                 DayName = DayNum & "Day"
47                 With ActiveSheet.Shapes(DayName)
48                     .Fill.ForeColor.RGB = RGB(255, 255, 255)
49                     .TextFrame2.TextRange.Font.Bold = msoFalse
50                 End With
51             Next DayNum
52         End If
53
54         Sheets("CalPopUp" ).Range("A3" ).Value = Month(Sheets("CalPopUp" ).Range("A1" ).
           Value) 'Set Month
55         Sheets("CalPopUp" ).Range("A2" ).Value = Year(Sheets("CalPopUp" ).Range("A1" ).
           Value) 'Set Year
56         DayName = Sheets("CalPopUp" ).Range("A20" ).Value & "Day"
57         ' UnGroupCal
58         If InStr(.Shapes("1Day" ).DrawingObject.Formula, "]" ) <> 0 Then 'Run Workbook

```

1 2 3

```

1 2 3
Link Remover and Cell Link Replacement
59 MacroLinkRemover
60 CalFormulaReplacement
61 End If
62 'GroupCal
63 On Error GoTo NoCal
64 If DayName = "Day" Then DayName = Day(Date) & "Day" 'Set Default Day
65 With ActiveSheet.Shapes(DayName)
66 .Fill.ForeColor.RGB = RGB(252, 213, 180)
67 .TextFrame2.TextRange.Font.Bold = msoTrue
68 End With
69 On Error GoTo NoCal
70 .Shapes("Calendar").Visible = msoCTrue
71 .Shapes.Range(Array("Settings", "CalCol1", "CalCol2", "CalCol3", "CalCol4",
"CalCol5", "CalCol6", "CalCol7", "CalCol8", "CalCol9")).Visible = False '
72 .Shapes("Calendar").Left = SelCell.Left - 48
73 .Shapes("Calendar").Placement = xlMove
74 .Shapes("Calendar").ZOrder msoBringToFront
75 .Shapes("Calendar").Top = SelCell.Offset(1, 0).Top
76 If Sheets("CalPopUp").Range("A6").Value > 0 Then
77 .Shapes.Range(Array("36Day", "37Day", "38Day", "39Day", "40Day", "41Day",
"42Day")).Visible = True
78 Else:
79 .Shapes.Range(Array("36Day", "37Day", "38Day", "39Day", "40Day", "41Day",
"42Day")).Visible = False
80 End If
81 Sheets("CalPopUp").Range("A7").Value = SelCell.Address
82 ActiveCell.Select
83 End With
84 Exit Sub
85 NoCal:
86 MsgBox "The Pop-up Calendar does not exist on this worksheet. Please copy the
calendar over from another sheet and paste into this sheet"
87 End Sub
88 Sub CalFormulaReplacement()
89 With ActiveSheet
90 Dim DayNum, ColNum, RowNum As Long
91 Dim Shp As Shape
92 ColNum = 2
93 RowNum = 1
94 For DayNum = 1 To 42
95 .Shapes(DayNum & "Day").DrawingObject.Formula = "=CalPopUp!" & .Cells(RowNum
, ColNum).Address
96 ColNum = ColNum + 1
97 If ColNum = 9 Then
98 ColNum = 2
99 RowNum = RowNum + 1
100 End If
101 Next DayNum
102 .Shapes("Month").DrawingObject.Formula = "=CalPopUp!$A$4"
103 .Shapes("Year").DrawingObject.Formula = "=CalPopUp!$A$2"
104 End With
105 End Sub
106
107 Sub CheckForSheet()
108 'Checks for existance of Calendar Pop-up Worksheet
109 Dim ws As Worksheet
110 On Error GoTo CreateWS
111 Set ws = ActiveWorkbook.Sheets("CalPopUp")
112 Exit Sub
113 CreateWS:

```

```

1
114 CreateCalSht
115 End Sub
116
117
118 'Create Calendar Sheet on First Run of Calendar
119 Sub CreateCalSht()
120 Dim ColCnt, RowCnt, DayCnt, CalCol As Long
121 Dim ws, ActSht As Worksheet
122 Set ActSht = ActiveSheet
123 'On Error GoTo NoCal
124 ActiveSheet.Shapes("Calendar").Copy
125 Set ws = ThisWorkbook.Sheets.Add(After:=ThisWorkbook.Sheets(ThisWorkbook.Sheets.
Count))
126 ws.Name = "CalPopUp"
127 ActSht.Activate
128
129 'Reassign Shape Links & Macros
130 With ActiveSheet
131 UnGroupCal
132 .Unprotect
133 .Shapes("PrevYr").OnAction = "" & ActiveWorkbook.Name & "!PrevYear"
134 .Shapes("NextYr").OnAction = "" & ActiveWorkbook.Name & "!NextYear"
135 .Shapes("NextRec").OnAction = "" & ActiveWorkbook.Name & "!NextMonth"
136 .Shapes("NextTri").OnAction = "" & ActiveWorkbook.Name & "!NextMonth"
137 .Shapes("PrevRec").OnAction = "" & ActiveWorkbook.Name & "!PrevMonth"
138 .Shapes("PrevTri").OnAction = "" & ActiveWorkbook.Name & "!PrevMonth"
139 .Shapes("SetBtn").OnAction = "" & ActiveWorkbook.Name & "!ShowSettings"
140 .Shapes("Month").DrawingObject.Formula = "=CalPopUp!A4"
141 .Shapes("Year").DrawingObject.Formula = "=CalPopUp!A2"
142 DayCnt = 1
143 For RowCnt = 1 To 6
144 For ColCnt = 2 To 8
145 .Shapes(DayCnt & "Day").DrawingObject.Formula = "=CalPopUp!" & .Cells(
RowCnt, ColCnt).Address 'Assigned Linked Cell
146 .Shapes(DayCnt & "Day").OnAction = "" & ActiveWorkbook.Name &
"!SelectDay" 'Assign Macro
147 DayCnt = DayCnt + 1
148 Next ColCnt
149 Next RowCnt
150
151 'Assign Color Macros
152 For CalCol = 1 To 9
153 .Shapes("CalCol" & CalCol).OnAction = "" & ActiveWorkbook.Name &
"!CalCol" 'Assign Color Macro
154 Next CalCol
155 End With
156
157 With Sheets("CalPopUp")
158 .Paste
159 .Visible = xlSheetHidden
160
161 'Add in Formulas and Details
162 .Range("A1").Value = Date 'Set Current Date
163 .Range("A2").Value = Year(Date) 'Set Current Year
164 .Range("A3").Value = Month(Date) 'Set Current Month #
165 .Range("A4").Value = "=INDEX(CalMonths,A3,)"
166 .Range("A5").Value = "=A4&" & Chr(34) & " " & Chr(34) & "&CalYear"
167 .Range("A6").Value = "=SUM(B6:H6)"
168 .Range("A8").Value = "January"
169 .Range("A8").AutoFill Destination:=.Range("A8:A19"), Type:=xlFillDefault
170 .Range("A20").Value =

```

1 2

```

1 2
"=IFERROR(INDIRECT(ADDRESS(SUMPRODUCT((B1:H6=A1)*ROW(B1:H6))+6,SUMPRODUCT((B1:H6=A
171 1)*COLUMN(B1:H6)),1,1)), " & Chr(34) & Chr(34) & ")
'Set Defined Names
172 ActiveWorkbook.Names.Add Name:="CalMonths" , RefersTo:="=CalPopUp!$A$8:$A$19"
173 ActiveWorkbook.Names.Add Name:="CalYear" , RefersTo:="=CalPopUp!$A$2"
174
175 'Add in Calendar Formulas
176
177 .Range("B1" ).Value =
"=IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=1,DATE(CalYear,MATCH($A$4,Ca
178 lMonths,0),1)," & Chr(34) & Chr(34) & ")
.Range("C1" ).Value = "=IF(B1<>" & Chr(34) & Chr(34) &
",B1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=2,DATE(CalYear,MATCH($A
179 $4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")")
.Range("D1" ).Value = "=IF(C1<>" & Chr(34) & Chr(34) &
",C1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=3,DATE(CalYear,MATCH($A
180 $4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")")
.Range("E1" ).Value = "=IF(D1<>" & Chr(34) & Chr(34) &
",D1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=4,DATE(CalYear,MATCH($A
181 $4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")")
.Range("F1" ).Value = "=IF(E1<>" & Chr(34) & Chr(34) &
",E1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=5,DATE(CalYear,MATCH($A
182 $4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")")
.Range("G1" ).Value = "=IF(F1<>" & Chr(34) & Chr(34) &
",F1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=6,DATE(CalYear,MATCH($A
183 $4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")")
.Range("H1" ).Value = "=IF(G1<>" & Chr(34) & Chr(34) &
",G1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=7,DATE(CalYear,MATCH($A
184 $4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")")
.Range("B2" ).Value = "=H1+1"
185 .Range("C2" ).Value = "=B2+1"
186 .Range("C2" ).AutoFill Destination:=.Range("C2:H2" ), Type:=xlFillDefault
187 .Range("B2:H2" ).AutoFill Destination:=.Range("B2:H4" ), Type:=xlFillDefault
188 .Range("B5" ).Value = "=IF(OR(H4=" & Chr(34) & Chr(34) & ",MONTH(H4+1)<>$A$3),"
& Chr(34) & Chr(34) & ",H4+1)"
189 .Range("C5" ).Value = "=IFERROR(IF(MONTH(B5+1)<>$A$3," & Chr(34) & Chr(34) &
",B5+1)," & Chr(34) & Chr(34) & ")")
190 .Range("B6" ).Value = "=IFERROR(IF(OR(H5=" & Chr(34) & Chr(34) &
",MONTH(H5+1)<>$A$3)," & Chr(34) & Chr(34) & ",H5+1)," & Chr(34) & Chr(34) &
")")
191 .Range("C6" ).Value = "=IFERROR(F(MONTH(I5+1)<>$A$3," & Chr(34) & Chr(34) &
",I5+1)," & Chr(34) & Chr(34) & ")")
192 .Range("C5:C6" ).AutoFill Destination:=.Range("C5:H6" ), Type:=xlFillDefault
193
194 'Set format to Single Day
195 .Range("B1:H6" ).NumberFormat = "d"
196
197 'Add in relative Day #'s
198 .Range("B7" ).Value = "1"
199 .Range("C7" ).Value = "2"
200 .Range("B8" ).Value = "8"
201 .Range("C8" ).Value = "9"
202 .Range("B7:C8" ).AutoFill Destination:=.Range("B7:H8" ), Type:=xlFillDefault
203 .Range("B7:H8" ).AutoFill Destination:=.Range("B7:H12" ), Type:=xlFillDefault
204 GroupCal
205 End With
206 Exit Sub
207 NoCal:
208 MsgBox "The Pop-up Calendar does not exist on this worksheet. Please copy the
calendar over from another sheet and paste into this sheet"
209 End Sub

```



```

210 Sub GroupCal()
211     ActiveSheet.Shapes.Range(Array("NextTri" , "NextRec" )).Group.Select
212     Selection.ShapeRange.Name = "NextMonth"
213     ActiveSheet.Shapes.Range(Array("PrevTri" , "PrevRec" )).Group.Select
214     Selection.ShapeRange.Name = "PrevMonth"
215     ActiveSheet.Shapes.Range(Array("Settings" , "40Day" , "41Day" , "39Day" , "38Day" _
216         , "42Day" , "37Day" , "36Day" , "CalBack" , "Month" , "Year" , "CalBorder" ,
217         "1Day" , _
218         "3Day" , "14Day" , "7Day" , "4Day" , "2Day" , "5Day" , "8Day" , "10Day" ,
219         "6Day" , _
220         "13Day" , "11Day" , "9Day" , "12Day" , "15Day" , "17Day" , "20Day" , "21Day" ,
221         "18Day" _
222         , "16Day" , "19Day" , "22Day" , "24Day" , "26Day" , "27Day" , "25Day" , "23Day"
223         , _
224         "28Day" , "29Day" , "31Day" , "34Day" , "35Day" , "32Day" , "30Day" , "33Day" ,
225         "Sa" , _
226         "Fr" , "Th" , "We" , "Tu" , "Mo" , "Su" , "SetBtn" , "CalCol1" , "CalCol2" ,
227         "CalCol3" , _
228         "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" ,
229         "PrevMonth" , _
230         "NextMonth" , "NextYr" , "PrevYr" )).Visible = msoCTrue
231     ActiveSheet.Shapes.Range(Array("Settings" , "40Day" , "41Day" , "39Day" , "38Day" _
232         , "42Day" , "37Day" , "36Day" , "CalBack" , "Month" , "Year" , "CalBorder" , "1Day"
233         , _
234         "3Day" , "14Day" , "7Day" , "4Day" , "2Day" , "5Day" , "8Day" , "10Day" , "6Day" , _
235         "13Day" , "11Day" , "9Day" , "12Day" , "15Day" , "17Day" , "20Day" , "21Day" ,
236         "18Day" _
237         , "16Day" , "19Day" , "22Day" , "24Day" , "26Day" , "27Day" , "25Day" , "23Day" , _
238         "28Day" , "29Day" , "31Day" , "34Day" , "35Day" , "32Day" , "30Day" , "33Day" ,
239         "Sa" , _
240         "Fr" , "Th" , "We" , "Tu" , "Mo" , "Su" , "SetBtn" , "CalCol1" , "CalCol2" ,
241         "CalCol3" , _
242         "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" , "PrevMonth"
243         , _
244         "NextMonth" , "NextYr" , "PrevYr" )).Select
245     Selection.ShapeRange.Group.Select
246     Selection.ShapeRange.Name = "Calendar"
247     Selection.Name = "Calendar"
248     Selection.Placement = xlMove
249     ActiveSheet.Shapes("Calendar" ).Placement = 2
250 End Sub

```

```

251
252
253
254
255 Sub MacroLinkRemover()
256     'PURPOSE: Remove an external workbook reference from all shapes triggering macros
257     'Source: www.ExcelForFreelancers.com
258     Dim Shp As Shape
259     Dim MacroLink, NewLink As String
260     Dim SplitLink As Variant
261
262     For Each Shp In ActiveSheet.Shapes 'Loop through each shape in worksheet
263         'Grab current macro link (if available)
264         On Error GoTo NextShp
265         MacroLink = Shp.OnAction

```

```

1 2
256 'Determine if shape was linking to a macro
257 If MacroLink <> "" And InStr(MacroLink, "!") <> 0 Then
258     'Split Macro Link at the exclamation mark (store in Array)
259     SplitLink = Split(MacroLink, "!")
260
261     'Pull text occurring after exclamation mark
262     NewLink = SplitLink(1)
263
264     'Remove any stragglng apostrophes from workbook name
265     If Right(NewLink, 1) = "'" Then
266         NewLink = Left(NewLink, Len(NewLink) - 1)
267     End If
268
269     'Apply New Link
270     Shp.OnAction = NewLink
271 End If
272 NextShp:
273 Next Shp
274 End Sub
275 Sub NextMonth()
276     'Next Month button
277     If Sheets("CalPopUp").Range("A20").Value <> Empty Then
278         DayName = Sheets("CalPopUp").Range("A20").Value & "Day"
279         With ActiveSheet.Shapes(DayName)
280             .Fill.ForeColor.RGB = RGB(255, 255, 255)
281             .TextFrame2.TextRange.Font.Bold = msoFalse
282         End With
283     End If
284     With Sheets("CalPopUp")
285         If .Range("A3").Value = 12 Then
286             .Range("A3").Value = 1
287             .Range("A2").Value = .Range("A2").Value + 1
288         Else:
289             .Range("A3").Value = .Range("A3").Value + 1
290         End If
291         If .Range("A6").Value > 0 Then
292             ActiveSheet.Shapes.Range(Array("36Day", "37Day", "38Day", "39Day",
293                 "40Day", "41Day", "42Day")).Visible = True
294         Else:
295             ActiveSheet.Shapes.Range(Array("36Day", "37Day", "38Day", "39Day",
296                 "40Day", "41Day", "42Day")).Visible = False
297         End If
298     End With
299 End Sub
300 Sub NextYear()
301     ThisWorkbook.Sheets("CalPopUp").Range("A2").Value = ThisWorkbook.Sheets(
302         "CalPopUp").Range("A2").Value + 1
303 End Sub
304 Sub PrevMonth()
305     'Previous Month Button
306     If Sheets("CalPopUp").Range("A20").Value <> Empty Then
307         DayName = Sheets("CalPopUp").Range("A20").Value & "Day"
308         With ActiveSheet.Shapes(DayName)
309             .Fill.ForeColor.RGB = RGB(255, 255, 255)
310             .TextFrame2.TextRange.Font.Bold = msoFalse
311         End With
312     End If
313     With Sheets("CalPopUp")
314         If .Range("A3").Value = 1 Then
315             .Range("A3").Value = 12
316         End If
317     End With
318 End Sub
1 2 3

```

```

1 2 3
314     .Range("A2" ).Value = .Range("A2" ).Value - 1
315     Else:
316     .Range("A3" ).Value = .Range("A3" ).Value - 1
317     End If
318     If .Range("A6" ).Value > 0 Then
319     ActiveSheet.Shapes.Range(Array("36Day" , "37Day" , "38Day" , "39Day" ,
    "40Day" , "41Day" , "42Day" )).Visible = True
320     Else:
321     ActiveSheet.Shapes.Range(Array("36Day" , "37Day" , "38Day" , "39Day" ,
    "40Day" , "41Day" , "42Day" )).Visible = False
322     End If
323     End With
324     End Sub
325
326     Sub PrevYear()
327     ThisWorkbook.Sheets("CalPopUp" ).Range("A2" ).Value = ThisWorkbook.Sheets(
    "CalPopUp" ).Range("A2" ).Value - 1
328     End Sub
329     Sub ReplaceCalendar() 'Shape Deleted
330     ThisWorkbook.Sheets("CalPopUp" ).Shapes("Calendar" ).Copy 'Copy From Developers
    sheet
331     ActiveCell.Select 'Select the active Cell
332     ActiveSheet.Paste 'Paste in Calendar
333     End Sub
334
335     .....
336     ''''Select Day Of The Month
337     .....
338     Sub SelectDay()
339     Dim DayNumb As Long, RowNumb As Long, ColNumb As Long
340     DayNumb = Replace(Application.Caller, "Day" , "" )
341     RowNumb = Application.WorksheetFunction.RoundUp(DayNumb / 7, 0)
342     ColNumb = DayNumb Mod 7 + 1
343     If ColNumb = 1 Then ColNumb = 8
344     'On Error Resume Next
345     If ThisWorkbook.Sheets("CalPopUp" ).Range("A7" ).Value = Empty Then Exit Sub
346     ActiveSheet.Range(ThisWorkbook.Sheets("CalPopUp" ).Range("A7" ).Value).Value =
    ThisWorkbook.Sheets("CalPopUp" ).Cells(RowNumb, ColNumb).Value
347     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
348     ActiveCell.Offset(0, 1).Select
349     End Sub
350     Sub ShowSettings()
351     'Show or Hide Calendar Settings Panel
352     If ActiveSheet.Shapes.Range(Array("Settings" )).Visible = True Then
353     ActiveSheet.Shapes.Range(Array("Settings" , "CalCol1" , "CalCol2" , "CalCol3" ,
    "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" )).Visible
    = False
354     Else:
355     ActiveSheet.Shapes.Range(Array("Settings" , "CalCol1" , "CalCol2" , "CalCol3" ,
    "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" )).Visible
    = True
356     End If
357     End Sub
358
359     Sub UngroupCal()
360     On Error Resume Next
361     ActiveSheet.Shapes("Calendar" ).Ungroup
362     ActiveSheet.Shapes("NextMonth" ).Ungroup
363     ActiveSheet.Shapes("PrevMonth" ).Ungroup
364     On Error GoTo 0
365     End Sub

```

- _, 13
- A**
Activate, 11
ActiveCell, 9, 10, 15
ActiveSheet, 9-11, 13-15
ActiveWorkbook, 10-12
ActSht, 11
Add, 11, 12
Address, 10, 11
AdvancedFilter, 7
After, 11
Application, 7-9, 15
Array, 9, 10, 13-15
AutoFill, 11, 12
- B**
Bold, 9, 10, 14
- C**
Cal, 3
CalCol, 9, 11
Calculation, 7, 8
CalendarHide, 3, 9
CalendarShow, 3, 9
CalFormulaReplacement, 10
Caller, 9, 15
Cells, 7, 10, 11, 15
CheckForSheet, 3, 10
Chr, 11, 12
ColCnt, 11
ColNum, 10
ColNumb, 15
Copy, 11, 15
CopyToRange, 7
Count, 11
CountLarge, 3
CreateCalSht, 11
CreateWS, 10
CriteriaRange, 7
- D**
Dashboard, 7, 8
Dashboard_Refresh, 3, 7, 8
Day, 10
DayCnt, 11
DayName, 9, 10, 14
DayNum, 9, 10
DayNumb, 15
Delete, 7
Destination, 11, 12
DrawingObject, 9-11
Duplicate, 7
- E**
Empty, 3, 9, 14, 15
Explicit, 2-7
- F**
Fill, 9, 10, 14
Font, 9, 10, 14
ForeColor, 9, 10, 14
Format, 7, 8
Formula, 9-11
- G**
Group, 13
GroupCal, 12, 13
- H**
Height, 7, 8
- I**
InStr, 7, 9, 14
Intersect, 3
IsDate, 9
- L**
LastResultRow, 7
LastRow, 7
Left, 7, 8, 10, 14
Len, 14
- M**
MacroLink, 13, 14
MacroLinkRemover, 10, 13
Month, 9, 11
MsgBox, 10, 12
msoBringToFront, 10
msoCTrue, 10, 13
msoFalse, 3, 9, 14, 15
msoTrue, 10
- N**
Name, 7, 11-13
Names, 12
NewLink, 13, 14
Next_Week, 8
NextDay, 7
NextMonth, 14
NextShp, 13, 14
NextYear, 14
NoCal, 9, 10, 12
NumberFormat, 12
- O**
Offset, 10, 15
OnAction, 11, 13, 14
- P**
Paste, 11, 15
Placement, 10, 13
Prev_Week, 8
PrevMonth, 14
PrevYear, 15
- R**
Range, 3, 7-15
RefersTo, 12
Replace, 15
ReplaceCalendar, 3, 15
ResultRow, 7
RGB, 9, 10, 14
Right, 14
RoundUp, 15
Row, 7
RowCnt, 11
RowNum, 10
RowNumb, 15
- S**
ScreenUpdating, 7, 8
SelCell, 9, 10
SelectDay, 15
Selection, 9, 13
Shape, 3, 7, 10, 13
ShapeRange, 9, 13
Shapes, 3, 7-11, 13-15
- Sheets, 9-11, 14, 15
ShowSettings, 15
Shp, 10, 13, 14
Split, 14
SplitLink, 13, 14
- T**
Target, 3
Text, 7, 8
TextFrame2, 7-10, 14
TextRange, 7-10, 14
ThisWorkbook, 11, 14, 15
Top, 7, 8, 10
TransAcct, 7
Transactions, 7, 8
TransAmt, 7
TransID, 7
TransNotes, 7
TransShp, 7
TransType, 7
- U**
Ungroup, 15
UnGroupCal, 11, 15
Unique, 7
- V**
Value, 3, 7-12, 14, 15
Visible, 3, 9-11, 13-15
- W**
WeekCol, 7
WeekRow, 7
Width, 7, 8
Worksheet, 10, 11
Worksheet_Change, 3
Worksheet_SelectionChange, 3
WorksheetFunction, 15
ws, 10, 11
- X**
xlCalculationAutomatic, 8
xlCalculationManual, 7
xlFillDefault, 11, 12
xlFilterCopy, 7
xlMove, 10, 13
xlSheetHidden, 11
xlUp, 7
- Y**
Year, 9, 11
- Z**
ZOrder, 10

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