

VBA SOURCE CODE BOOK



**Top 5 Excel Hacks: Learn To
Build Applications fast**



**DOWNLOAD
APPLICATION**



**VIEW
TRAINING**

by: Randy Austin

ABOUT THE AUTHOR

A 4-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 471,000 YouTube subscribers, 35,448,742 video views, 430+ comprehensive training videos, and a thriving 65,000 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](#)



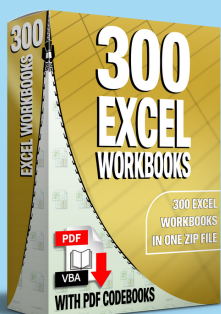
16 hour masterclass that will teach you the tips, tricks and techniques on how to create a dynamic single-click dashboard, and a ton more

[Click here to learn more](#)



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.

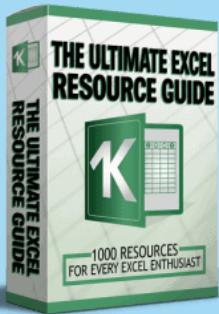
[Click here to learn more](#)



Incredible Package of 300 of my BEST Applications with PDF VBA Codebooks packed into a SINGLE ZIP File which also includes the "300 Workbook Library".

[Click here to learn more](#)

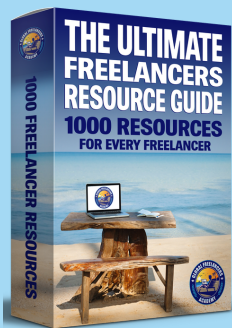
OUR COURSES & PRODUCTS



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](#)

-



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](#)

-



Revolutionize the way you work with Excel and take productivity to the next level with the Excel AI Toolpack - the FIRST AI tool designed for ANY Windows Desktop version of Excel. This incredible add-in combines FIVE powerful AI tools, transforming your Excel into an intelligent powerhouse!

[Click here to learn more](#)

Projects.....	3
VBAProject.....	3
Documents.....	3
Admin.....	3
(Declarations).....	3
Worksheet_SelectionChange [Sub].....	3
ApptsDB.....	4
(Declarations).....	4
Schedule.....	5
(Declarations).....	5
Worksheet_SelectionChange [Sub].....	5
Sheet1.....	6
(Declarations).....	6
Sheet2.....	7
(Declarations).....	7
ThisWorkbook.....	8
(Declarations).....	8
Forms.....	9
ContactFrm.....	9
(Declarations).....	9
CancelBtn_Click [Sub].....	9
SaveBtn_Click [Sub].....	9
Modules.....	10
Admin_Macros.....	10
(Declarations).....	10
Admin_SetPlatformColor [Sub].....	10
Appt_Macros.....	11
(Declarations).....	11
Appt_Delete [Sub].....	11
Appt_Load [Sub].....	11
Appt_New [Sub].....	11
Appt_SaveUpdate [Sub].....	11
Contact_Macros.....	12
(Declarations).....	12
Contact_AddNew [Sub].....	12
Contact_Delete [Sub].....	12
Contact_Edit [Sub].....	12
Contact_SaveUpdate [Sub].....	12
Contact_SortNames [Sub].....	13
PopUp_Calendar.....	14
(Declarations).....	14
CalCol [Sub].....	14
CalendarHide [Sub].....	14
CalendarShow [Sub].....	14
CalFormulaReplacement [Sub].....	15
CheckForSheet [Sub].....	15
CreateCalSht [Sub].....	16
GroupCal [Sub].....	18
MacroLinkRemover [Sub].....	18
NextMonth [Sub].....	19
NextYear [Sub].....	19
PrevMonth [Sub].....	19
PrevYear [Sub].....	20
ReplaceCalendar [Sub].....	20
SelectDay [Sub].....	20
ShowSettings [Sub].....	20
UnGroupCal [Sub].....	20
Recurring_Macros.....	21
(Declarations).....	21
CreateRecurringAppt [Sub].....	21
Schedule_Macros.....	22
(Declarations).....	22
PrintSchedule [Sub].....	22
Schedule_Appt_Select [Sub].....	22
Schedule_CheckForMove [Sub].....	22
Schedule_NextMonth [Sub].....	23

Table of Contents

Schedule_PrevMonth [Sub]	23
Schedule_Refresh [Sub]	23
Schedule_ThisMonth [Sub]	25
SendToOutlook_Macros	26
(Declarations)	26
SendAppointmentToOutlook [Sub]	26

```
1 Option Explicit
2
3 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
4     If Target.CountLarge > 1 Then Exit Sub
5     If Shapes("ColorPalette").Visible = True Then Shapes("ColorPalette").Visible =
        msoFalse
6
7     'Show color palette
8     If Not Intersect(Target, Range("F7,F9")) Is Nothing Then
9         With Shapes("ColorPalette")
10            .Left = Range("F" & Target.Row).Left
11            .Top = Range("F" & Target.Row + 1).Top
12            .Visible = msoCTrue
13        End With
14    End If
15 End Sub
```

1
2

Option Explicit


```
1 Option Explicit
2
3 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
4     'Check To make sure calendar exists, if not, copy from Developers sheet and paste
      it in
5     Dim Cal As Shape
6     On Error Resume Next
7     Set Cal = Shapes("Calendar" )
8     On Error GoTo 0
9     If Cal Is Nothing Then ReplaceCalendar 'Shape Deleted
10    If Shapes("Calendar" ).Visible = True Then CalendarHide 'Run macro to hide calendar
11
12    If Target.Count > 1 Then Exit Sub
13    If Not Intersect(Target, Range("M6,M14" )) Is Nothing Then 'Change this to any cell
      (or cells) you would like to have the Pop-Up Calendar Appear
14        CheckForSheet
15        CalendarShow
16    Else:
17        CheckForSheet
18        CalendarHide
19    End If
20 End Sub
```

1 Option Explicit

2

1
2

Option Explicit

1 Option Explicit

2

```
1  
2 Option Explicit  
3  
4  
5 Private Sub CancelBtn_Click()  
6     Unload Me  
7 End Sub  
8  
9 Private Sub SaveBtn_Click()  
10    Contact_SaveUpdate  
11 End Sub  
12  
13
```

1 Option Explicit

2

3 Sub Admin_SetPlatformColor()

4 ActiveCell.Interior.Color = Admin.Shapes(Application.Caller).Fill.ForeColor.RGB

5 Admin.Shapes("ColorPalette").Visible = msoFalse

6 End Sub

```

1 Option Explicit
2
3 Dim ApptRow As Long, ApptCol As Long
4
5 Sub Appt_Delete()
6     If MsgBox("Are you sure you want to delete this Schedule Appointment?" , vbYesNo,
7         "Delete Appt" ) = vbNo Then Exit Sub
8     With Schedule
9         .Range("B1" ).Value = True 'Exit out of Check For Change Loop
10        If .Range("B10" ).Value = Empty Then GoTo NotSaved
11        ApptRow = .Range("B10" ).Value 'Appt row
12        ApptsDB.Range(ApptRow & ":" & ApptRow).EntireRow.Delete 'Delete Row
13        NotSaved:
14        Appt_New
15        Schedule_Refresh 'Refresh Schedule
16    End With
17 End Sub
18
19 Sub Appt_Load()
20     With Schedule
21         If .Range("B10" ).Value = Empty Then
22             MsgBox "Please select a correct Schedule Appt to load"
23             Exit Sub
24         End If
25         ApptRow = .Range("B10" ).Value 'Appt Row
26         For ApptCol = 2 To 7
27             .Range("M" & ApptCol + 2).Value = ApptsDB.Cells(ApptRow, ApptCol).Value
28             'Bring Over data
29         Next ApptCol
30     End With
31 End Sub
32
33 Sub Appt_New()
34     Schedule.Range("B9,M4:M9" ).ClearContents 'Clear Out
35 End Sub
36
37 Sub Appt_SaveUpdate()
38     With Schedule
39         .Range("B1" ).Value = True 'Exit out of Check For Change Loop
40         If Application.WorksheetFunction.CountA(.Range("M4:M6" )) < 3 Then
41             MsgBox "Please make sure to add in an Appt Name, Date & Time"
42             Exit Sub
43         End If
44         If .Range("B10" ).Value = Empty Then 'New Appt
45             ApptRow = ApptsDB.Range("A99999" ).End(xlUp).Row + 1 'First Avail. Row
46             .Range("B9" ).Value = .Range("B11" ).Value 'Set Appt ID
47             ApptsDB.Range("A" & ApptRow).Value = .Range("B9" ).Value 'Appt ID
48         Else 'Existing Appt
49             ApptRow = .Range("B10" ).Value 'Existing Appt Row
50         End If
51         For ApptCol = 2 To 7
52             ApptsDB.Cells(ApptRow, ApptCol).Value = .Range("M" & ApptCol + 2).Value
53             'Save/ Update data
54         Next ApptCol
55         If .Range("B12" ).Value = False Then 'Refresh & Set Saved Message except for
56             recurring
57             Schedule_Refresh 'Refresh Schedule
58             MsgBox "Appointment Saved"
59         End If
60     End With
61 End Sub

```

```

1 Option Explicit
2 Dim ContactRow As Long, ContactCol As Long, LastRow As Long, LastResultRow As Long
3 Dim ContactFld As control
4
5 Sub Contact_AddNew()
6     Schedule.Range("M5" ).ClearContents 'Clear Selected Contact Row and Contact DB Row
7     ContactFrm.Show
8 End Sub
9
10
11 Sub Contact_Delete()
12     If MsgBox("Are you sure you want to delete this Contact?" , vbYesNo, "Delete
13     Contact" ) = vbNo Then Exit Sub
14     If Schedule.Range("B13" ).Value = Empty Then Exit Sub
15     ContactRow = Schedule.Range("B13" ).Value 'Contact Row
16     For ContactCol = 2 To 9
17         ContactsDB.Cells(ContactRow, ContactCol).ClearContents
18     Next ContactCol
19     Schedule.Range("M5" ).ClearContents
20     Contact_SortNames
21 End Sub
22
23
24
25 Sub Contact_Edit()
26     With Schedule
27         If .Range("B13" ).Value = Empty Then
28             MsgBox "Please select on a Contact to edit"
29             Exit Sub
30         End If
31         ContactRow = .Range("B13" ).Value 'Contact Row
32         With ContactFrm
33             For ContactCol = 2 To 8
34                 Set ContactFld = .Controls("Field" & ContactCol - 1)
35                 ContactFld.Value = ContactsDB.Cells(ContactRow, ContactCol).Value 'Add
36             Next ContactCol
37             .Show 'Display User Form
38         End With
39     End With
40 End Sub
41
42
43 Sub Contact_SaveUpdate()
44     'Check For Required field
45     With ContactFrm
46         If .Field1.Value = Empty Then
47             MsgBox "Please make sure to add a Contact Name before saving"
48             Exit Sub
49         End If
50         If Schedule.Range("B13" ).Value = Empty Then 'New Contact
51             ContactRow = ContactsDB.Range("A9999" ).End(xlUp).Row + 1 'First Avail. Row
52             ContactsDB.Range("A" & ContactRow).Value = Schedule.Range("B14" ).Value
53             'Next Contact ID
54             ContactsDB.Range("I" & ContactRow).Value = "=Row()"
55         Else 'Existing Contact.
56             ContactRow = Schedule.Range("B13" ).Value 'Contact Row
57         End If
58         For ContactCol = 2 To 8
59             Set ContactFld = .Controls("Field" & ContactCol - 1)

```

1 2 3


```

1 2 3
59   ContactsDB.Cells(ContactRow, ContactCol).Value = ContactFld.Value
60   Next ContactCol
61   Schedule.Range("M5").Value = .Field1.Value 'Set Name (in case of changes)
62   End With
63   Unload ContactFrm
64   Contact_SortNames 'Resort Names
65 End Sub

66
67 Sub Contact_SortNames()
68   With ContactsDB
69     LastRow = .Range("A99999").End(xlUp).Row
70     If LastRow < 4 Then Exit Sub
71     On Error Resume Next
72     .Names("Criterial").Delete
73     On Error GoTo 0
74     .Range("B3:B" & LastRow).AdvancedFilter xlFilterCopy, , CopyToRange:=.Range(
75       "L2" ), Unique:=True
76     LastResultRow = .Range("L99999").End(xlUp).Row
77     If LastResultRow < 4 Then Exit Sub 'No Sort On 1 Row Of data
78     'Sort By Name & remove Blank Rows
79     With .Sort
80       .SortFields.Clear
81       .SortFields.Add Key:=ContactsDB.Range("L3" ), SortOn:=xlSortOnValues, Order:=
82         xlAscending, DataOption:=xlSortNormal 'Sort
83     .Apply 'Apply Sort
84   End With
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
32         CalPopUp Sheet
33 End Sub
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" ).
55         Value) 'Set Month
56         Sheets("CalPopUp" ).Range("A2" ).Value = Year(Sheets("CalPopUp" ).Range("A1" ).
57         Value) 'Set Year
58         DayName = Sheets("CalPopUp" ).Range("A20" ).Value & "Day"
59         ' UnGroupCal
60         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",
72 "CalCol5", "CalCol6", "CalCol7", "CalCol8", "CalCol9")).Visible = False '
73 .Shapes("Calendar").Left = SelCell.Left
74 .Shapes("Calendar").Placement = xlMove
75 .Shapes("Calendar").ZOrder msoBringToFront
76 .Shapes("Calendar").Top = SelCell.Offset(1, 0).Top
77 If Sheets("CalPopUp").Range("A6").Value > 0 Then
78     .Shapes.Range(Array("36Day", "37Day", "38Day", "39Day", "40Day", "41Day",
79 "42Day")).Visible = True
80 Else:
81     .Shapes.Range(Array("36Day", "37Day", "38Day", "39Day", "40Day", "41Day",
82 "42Day")).Visible = False
83 End If
84 Sheets("CalPopUp").Range("A7").Value = SelCell.Address
85 ActiveCell.Select
86 End With
87 Exit Sub
88 NoCal:
89 MsgBox "The Pop-up Calendar does not exist on this worksheet. Please copy the
90 calendar over from another sheet and paste into this sheet"
91 End Sub
92
93 Sub CalFormulaReplacement()
94 With ActiveSheet
95     Dim DayNum, ColNum, RowNum As Long
96     Dim Shp As Shape
97     ColNum = 2
98     RowNum = 1
99     For DayNum = 1 To 42
100         .Shapes(DayNum & "Day").DrawingObject.Formula = "=CalPopUp!" & .Cells(RowNum
101         , ColNum).Address
102         ColNum = ColNum + 1
103         If ColNum = 9 Then
104             ColNum = 2
105             RowNum = RowNum + 1
106         End If
107     Next DayNum
108     .Shapes("Month").DrawingObject.Formula = "=CalPopUp!$A$4"
109     .Shapes("Year").DrawingObject.Formula = "=CalPopUp!$A$2"
110 End With
111 End Sub
112
113 Sub CheckForSheet()
114 'Checks for existance of Calendar Pop-up Worksheet
115 Dim ws As Worksheet
116 On Error GoTo CreateWS
117 Set ws = ActiveWorkbook.Sheets("CalPopUp")
118 Exit Sub
119 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

```



```
1 Option Explicit
2
3 Sub CreateRecurringAppt()
4     Dim ApptDate As Date
5     Dim Freq As String, FreqQty As Long, ApptQty As Long, ApptNumb As Long
6     With Schedule
7         If .Range("B10").Value = Empty Then
8             MsgBox "Please make sure to save this Appt before creating recurring Schedule
9             Appt"
10            Exit Sub
11        End If
12        If Application.WorksheetFunction.CountA(.Range("M12:M15")) < 4 Then
13            MsgBox "Please make sure to fill in all recurring fields"
14            Exit Sub
15        End If
16        FreqQty = .Range("M12").Value 'Frequency Qty
17        Freq = .Range("M13").Value 'Frequency
18        ApptDate = .Range("M14").Value 'Start Date
19        ApptQty = .Range("M15").Value ' Number of Schedule Appts
20        .Range("B12").Value = True 'Set Recurring to True
21        For ApptNumb = 1 To ApptQty
22            .Range("M6").Value = ApptDate 'Cal. Appt Date
23            .Range("B9").ClearContents 'Clear Existing Appt ID
24            Appt_SaveUpdate 'Save Schedule Appt
25            Select Case Freq
26            Case Is = "Minute(s)"
27                ApptDate = DateAdd("n", FreqQty, ApptDate)
28            Case Is = "Hour(s)"
29                ApptDate = DateAdd("h", FreqQty, ApptDate)
30            Case Is = "Day(s)"
31                ApptDate = DateAdd("d", FreqQty, ApptDate)
32            Case Is = "Week(s)"
33                ApptDate = DateAdd("ww", FreqQty, ApptDate)
34            Case Is = "Month(s)"
35                ApptDate = DateAdd("m", FreqQty, Now)
36            End Select
37        Next ApptNumb
38        .Range("B12").Value = False 'Set Recurring to false
39        Schedule_Refresh 'Refresh Schedule
40        MsgBox ApptQty & " Schedule Appts have been created"
41    End With
42 End Sub
```

```

1 Option Explicit
2 Dim CalRow As Long, CalCol As Long, LastRow As Long, LastResultRow As Long, ResultRow
  As Long
3 Dim ApptNumb As Long, DayApptCount As Long, ApptTop As Long, NextDateRow As Long,
  ApptLimit As Long
4 Dim ApptColor As String, ApptID As String, ContName As String
5 Dim ApptShp As Shape
6 Dim ApptDate As Date, ApptTime As String
7 Dim ApptWidth As Double, ApptLeft As Double
8
9 Sub PrintSchedule()
10     Schedule.PrintOut , , , False, True, , , , False
11 End Sub
12
13
14 Sub Schedule_Appt_Select()
15     With Schedule
16         ApptID = Replace(Application.Caller, Left(Application.Caller, 7), "" ) 'Extract
          Appt ID
17         For Each ApptShp In .Shapes 'Reset All Colors back to default color
18             If InStr(ApptShp.Name, "CalAppt" ) > 0 Then ApptShp.Fill.ForeColor.RGB = Admin
              .Range("F7" ).Interior.Color
19         Next ApptShp
20         .Shapes("CalAppt" & ApptID).Fill.ForeColor.RGB = Admin.Range("F9" ).Interior.
          Color 'Select Appt. Color
21         .Range("B9" ).Value = ApptID
22         'Drag & Drop Settings
23         Schedule.Range("B9" ).Value = ApptID
24         .Range("B15" ).Value = Schedule.Shapes(Application.Caller).Left 'set Selected
          Left Position
25         .Range("B16" ).Value = Schedule.Shapes(Application.Caller).Top ' Set Selected
          Top Position
26         .Range("B1" ).Value = False 'Set Schedule Move to False
27         .Shapes("CalAppt" & ApptID).Select
28         Appt_Load
29         Schedule_CheckForMove
30     End With
31 End Sub
32
33
34 Sub Schedule_CheckForMove()
35     Dim DestRow As Long, DestCol As Long, ApptRow As Long, ApptCol As Long, CountDelay
      As Long
36     With Schedule
37         If .Range("B10" ).Value = Empty Then Exit Sub
38         ApptID = .Range("B9" ).Value 'Appt ID
39
40         For CountDelay = 1 To 100000
41             DoEvents
42             If .Range("B1" ).Value = True Then End 'Exit Loop on Move Appt Move = True
43             On Error Resume Next
44             With .Shapes("CalAppt" & ApptID)
45                 If .Left <> Schedule.Range("B15" ).Value Or .Top <> Schedule.Range("B16" ).
                  Value Then 'Move Detect
46                     'Check for incorrect move
47                     If .Left < Schedule.Range("D1" ).Left Or .Left > Schedule.Range("K1" ).
                      Left Or .Top > Schedule.Range("A39" ).Top Or .Top < Schedule.Range("A4"
                        ).Top - 1 Then
48                         MsgBox "Plase make sure to move the Schedule Appt to a correct
                          Schedule date on the Schedule"
49                         Schedule_Refresh

```

1 2 3 4 5 6

```

50     End
51     Exit Sub
52 End If
53
54     DestRow = Schedule.Shapes("CalAppt" & ApptID).TopLeftCell.Row 'Row
55     Destination
56     DestCol = Schedule.Shapes("CalAppt" & ApptID).TopLeftCell.Column
57     'Column Dest.
58     DestRow = DestRow - (DestRow + 2) Mod 6 'Date Row
59     ApptDate = Schedule.Cells(DestRow, DestCol).Value 'Schedule Date
60     If ApptDate = 0 Then
61         MsgBox "Please move the Appt to a day with an existing Date"
62         Schedule_Refresh
63         Exit Sub
64     End If
65     Schedule.Range("M6" ).Value = ApptDate 'new Appt dates
66     Appt_SaveUpdate 'Save Appt Changes
67     Schedule.Range("B1" ).Value = True 'Set Move Appt to true
68     End
69 End If
70 End With
71 Next CountDelay
72 .Range("B1" ).Value = True 'Set Move Appt Movement to True
73 End With
74 End Sub
75
76 Sub Schedule_NextMonth()
77     With Schedule
78         If .Range("B2" ).Value = 7 And .Range("B5" ).Value = 12 Then
79             MsgBox "You are at the last year and first last available"
80             Exit Sub
81         End If
82         If .Range("B5" ).Value = 12 Then 'December Month
83             .Range("B2" ).Value = .Range("B2" ).Value + 1 'Increase Year by 1
84             .Range("B5" ).Value = 1 'Set To January
85         Else 'Not January
86             .Range("B5" ).Value = .Range("B5" ).Value + 1 'Increase month by 1
87         End If
88         Schedule_Refresh 'Refresh Schedule
89     End With
90 End Sub
91
92 Sub Schedule_PrevMonth()
93     With Schedule
94         If .Range("B2" ).Value = 1 And .Range("B5" ).Value = 1 Then
95             MsgBox "You are at the first year and first month available"
96             Exit Sub
97         End If
98         If .Range("B5" ).Value = 1 Then 'January Month
99             .Range("B2" ).Value = .Range("B2" ).Value - 1 'Reduce Year by 1
100            .Range("B5" ).Value = 12 'Set To December
101        Else 'Not January
102            .Range("B5" ).Value = .Range("B5" ).Value - 1 'Reduce month by 1
103        End If
104        Schedule_Refresh 'Refresh Schedule
105    End With
106 End Sub
107
108 Sub Schedule_Refresh()
109     'Clear all existing Appt shapes from Schedule sheet
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

```

```

1
109 For Each ApptShp In Schedule.Shapes
110     If InStr(ApptShp.Name, "CalAppt" ) > 0 Then ApptShp.Delete
111 Next ApptShp
112 ApptColor = Admin.Range("F7" ).Interior.Color 'Set Appt Shape Color
113 ApptNumb = 1 'Set default Appt # to 1
114 ApptWidth = 1 'Set Default to 1
115 With ApptsDB
116     LastRow = .Range("A99999" ).End(xlUp).Row 'Last Appt Row
117     If LastRow < 4 Then Exit Sub
118     Application.ScreenUpdating = False
119     .Range("A3:F" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
120         "I2:J3" ), CopyToRange:=.Range("M2:Q2" ), Unique:=True
121     LastResultRow = .Range("M99999" ).End(xlUp).Row
122     If LastResultRow < 3 Then
123         Application.ScreenUpdating = True
124         Exit Sub
125     End If
126     If LastResultRow < 4 Then GoTo NoSort
127     With .Sort
128         .SortFields.Clear
129         .SortFields.Add Key:=ApptsDB.Range("O3" ), SortOn:=xlSortOnValues, Order:=
130             xlAscending, DataOption:=xlSortNormal 'Sort Based On Date
131         .SortFields.Add Key:=ApptsDB.Range("P3" ), SortOn:=xlSortOnValues, Order:=
132             xlAscending, DataOption:=xlSortNormal 'Sort Based On Time
133         .SetRange ApptsDB.Range("M3:Q" & LastResultRow) 'Set Range
134         .Apply 'Apply Sort
135     End With
136     NoSort:
137     For ResultRow = 3 To LastResultRow
138         ApptID = .Range("M" & ResultRow).Value 'Appt ID
139         ContName = .Range("N" & ResultRow).Value 'Appt Name
140         ApptDate = .Range("O" & ResultRow).Value 'Appt Date
141         ApptTime = Format(.Range("P" & ResultRow).Value, "h:mm/p" )
142         DayApptCount = Application.WorksheetFunction.CountIf([ItemDate_Results],
143             ApptDate) 'Get # of Appts in a single Date
144         If DayApptCount > 1 And ApptNumb = 1 Then ApptLimit = DayApptCount
145         If DayApptCount > 5 Then
146             If DayApptCount > 10 And ApptNumb = 1 Then 'On First Instance of Multiple
147                 any day count greater than 10
148                 NextDateRow = ResultRow + DayApptCount - 1 'Determine Next Day Result
149                 Row
150                 ApptLimit = 10 'Set Appt Limit to 10
151             End If
152             ApptWidth = 0.5 'Set Appt with to 1/2
153         End If
154         For CalRow = 4 To 34 Step 6
155             For CalCol = 4 To 10
156                 If Schedule.Cells(CalRow, CalCol).Value = ApptDate Then 'Day Found
157                     Schedule.Shapes("SampleApptShp" ).Duplicate.Name = "CalAppt" &
158                         ApptID
159                     With Schedule.Shapes("CalAppt" & ApptID)
160                         .Left = Schedule.Cells(CalRow, CalCol).Left + ApptLeft 'Set Left
161                         Pos.
162                         .Top = Schedule.Cells(CalRow + ApptNumb + ApptTop, CalCol).Top + 1
163
164                         .Width = Schedule.Cells(CalRow + ApptNumb, CalCol).Width *
165                             ApptWidth
166                         .Height = Schedule.Cells(CalRow + ApptNumb, CalCol).Height
167                         .TextFrame2.TextRange.Text = ApptTime & ":" & ContName 'Text
168                         inside shape
169                         .Fill.ForeColor.RGB = ApptColor 'Set Appt Color

```

1 2 3 4 5 6 7

```

159 | 1 2 3 4 5 6 7 | .OnAction = "Schedule_Appt_Select" 'Macro to select Appt
160 | | End With
161 | | If ApptNumb >= ApptLimit Then
162 | |     ApptNumb = 1 'Reset
163 | |     ApptWidth = 1 'Reset Appt Width
164 | |     ApptLimit = 1 'Reset Appt Limit
165 | |     If NextDateRow <> 0 Then
166 | |         ResultRow = NextDateRow
167 | |         NextDateRow = 0 'Reset Next Date Row to 0
168 | |     End If '
169 | | Else
170 | |     ApptNumb = ApptNumb + 1 'Increment by 1
171 | | End If
172 | | If ApptNumb <= 5 Then
173 | |     ApptLeft = 0 'Set Left Position
174 | |     ApptTop = 0
175 | | Else 'Appt Number from 6 to 10
176 | |     ApptLeft = Schedule.Cells(CalRow + ApptNumb, CalCol).Width / 2
177 | |     ApptTop = -5
178 | | End If
179 | | End If
180 | | Next CalCol
181 | | Next CalRow
182 | | Next ResultRow
183 | End With
184 | Application.ScreenUpdating = True
185 | End Sub
186 |
187 | Sub Schedule_ThisMonth()
188 |     With Schedule
189 |         .Range("B5" ).Value = Month(Date) 'Set Current Month #
190 |         .Range("B2" ).Value = Admin.Range("Years" ).Find(Year(Date), , xlFormulas,
191 |             xlWhole).Row - 3
192 |         Schedule_Refresh 'Refresh Schedule
193 |     End With
194 | End Sub

```

```
1 Option Explicit
2
3 Sub SendAppointmentToOutlook()
4     Dim ApptDate As Date, ApptTime As Date, ContName As String
5     Dim ApptName As String, Notes As String
6     Dim Dur As Double
7     With Schedule
8         ApptName = .Range("M4").Value 'Appt Name
9         ContName = .Range("M5").Value 'Contact Name
10        ApptDate = .Range("M6").Value 'Appt Date
11        ApptTime = .Range("M7").Value 'Time
12        Dur = .Range("M8").Value * 24 * 60 'Duration (in minutes)
13        Notes = .Range("M9").Value 'Notes
14    End With
15
16
17    Dim OutApp As Object, OutAppt As Object
18    Set OutApp = CreateObject("Outlook.Application")
19    Set OutAppt = OutApp.CreateItem(1)
20    With OutAppt
21        .Subject = ContName & ": " & ApptName
22        .Start = ApptDate + ApptTime
23        .Duration = Dur
24        .ReminderSet = True
25        .ReminderMinutesBeforeStart = 15
26        .Body = Notes
27        .Save
28        .Display
29    End With
30    Set OutApp = Nothing
31    Set OutAppt = Nothing
32    MsgBox "Schedule Appt has been sent to Outlook"
33 End Sub
```

—
_, 18

A

Activate, 16
ActiveCell, 10, 14, 15, 20
ActiveSheet, 14-16, 18-20
ActiveWorkbook, 15-17
ActSht, 16
Add, 13, 16, 17, 24
Address, 15, 16
Admin, 10, 22, 24, 25
Admin_SetPlatformColor, 10
AdvancedFilter, 13, 24
After, 16
Application, 10, 11, 14, 20-22, 24, 25
Apply, 13, 24
Appt_Delete, 11
Appt_Load, 11, 22
Appt_New, 11
Appt_SaveUpdate, 11, 21, 23
ApptCol, 11, 22
ApptColor, 22, 24
ApptDate, 21-24, 26
ApptID, 22-24
ApptLeft, 22, 24, 25
ApptLimit, 22, 24, 25
ApptName, 26
ApptNumb, 21, 22, 24, 25
ApptQty, 21
ApptRow, 11, 22
ApptsDB, 11, 24
ApptShp, 22, 24
ApptTime, 22, 24, 26
ApptTop, 22, 24, 25
ApptWidth, 22, 24, 25
Array, 14, 15, 18-20
AutoFill, 16, 17

B

Body, 26
Bold, 14, 15, 19

C

Cal, 5
CalCol, 14, 16, 22, 24, 25
CalendarHide, 5, 14
CalendarShow, 5, 14
CalFormulaReplacement, 15
Caller, 10, 14, 20, 22
CalRow, 22, 24, 25
CancelBtn_Click, 9
Cells, 11-13, 15, 16, 20, 23-25
CheckForSheet, 5, 15
Chr, 16, 17
Clear, 13, 24
ClearContents, 11, 12, 21
ColCnt, 16
ColNum, 15
ColNumb, 20
Color, 10, 22, 24
Column, 23
Contact_AddNew, 12
Contact_Delete, 12
Contact_Edit, 12
Contact_SaveUpdate, 9, 12
Contact_SortNames, 12, 13
ContactCol, 12, 13
ContactFld, 12, 13

ContactFrm, 12, 13
ContactRow, 12, 13
ContactsDB, 12, 13
ContName, 22, 24, 26
control, 12
Controls, 12
Copy, 16, 20
CopyToRange, 13, 24
Count, 5, 16
CountA, 11, 21
CountDelay, 22, 23
CountIf, 24
CountLarge, 3
CreateCalSht, 16
CreateItem, 26
CreateObject, 26
CreateRecurringAppt, 21
CreateWS, 15
CriteriaRange, 24

D

DataOption, 13, 24
DateAdd, 21
Day, 15
DayApptCount, 22, 24
DayCnt, 16
DayName, 14, 15, 19
DayNum, 14, 15
DayNumb, 20
Delete, 11, 13, 24
DestCol, 22, 23
Destination, 16, 17
DestRow, 22, 23
Display, 26
DoEvents, 22
DrawingObject, 14-16
Duplicate, 24
Dur, 26
Duration, 26

E

Empty, 11, 12, 14, 19-22
EntireRow, 11
Explicit, 3-12, 21, 22, 26

F

Field1, 12, 13
Fill, 10, 14, 15, 19, 22, 24
Find, 25
Font, 14, 15, 19
ForeColor, 10, 14, 15, 19, 22, 24
Format, 24
Formula, 14-16
Freq, 21
FreqQty, 21

G

Group, 18
GroupCal, 17, 18

H

Height, 24

I

InStr, 14, 19, 22, 24
Interior, 10, 22, 24
Intersect, 3, 5
IsDate, 14
ItemDate_Results, 24

K

Key, 13, 24

L

LastResultRow, 12, 13, 22, 24
LastRow, 12, 13, 22, 24
Left, 3, 15, 19, 22, 24
Len, 19

M

MacroLink, 18, 19
MacroLinkRemover, 15, 18
Month, 14, 16, 25
MsgBox, 11, 12, 15, 17, 21-23, 26
msoBringToFront, 15
msoCTrue, 3, 15, 18
msoFalse, 3, 10, 14, 19, 20
msoTrue, 15

N

Name, 16-18, 22, 24
Names, 13, 17
NewLink, 18, 19
NextDateRow, 22, 24, 25
NextMonth, 19
NextShp, 18, 19
NextYear, 19
NoCal, 14, 15, 17
NoSort, 24
Notes, 26
NotSaved, 11
Now, 21
NumberFormat, 17

O

Offset, 15, 20
OnAction, 16, 18, 19, 25
Order, 13, 24
OutApp, 26
OutAppt, 26

P

Paste, 16, 20
Placement, 15, 18
PrevMonth, 19
PrevYear, 20
PrintOut, 22
PrintSchedule, 22

R

Range, 3, 5, 11-26
RefersTo, 17
ReminderMinutesBeforeStart, 26
ReminderSet, 26
Replace, 20, 22
ReplaceCalendar, 5, 20
ResultRow, 22, 24, 25
RGB, 10, 14, 15, 19, 22, 24
Right, 19
RoundUp, 20
Row, 3, 11-13, 23-25
RowCnt, 16
RowNum, 15
RowNumb, 20

S

Save, 26
SaveBtn_Click, 9
Schedule, 11-13, 21-26

Schedule_Appt_Select, 22
Schedule_CheckForMove, 22
Schedule_NextMonth, 23
Schedule_PrevMonth, 23
Schedule_Refresh, 11, 21-23, 25
Schedule_ThisMonth, 25
ScreenUpdating, 24, 25
SelCell, 14, 15
SelectDay, 20
Selection, 14, 18
SendAppointmentToOutlook, 26
SetRange, 13, 24
Shape, 5, 15, 18, 22
ShapeRange, 14, 18
Shapes, 3, 5, 10, 14-16, 18-20, 22-24
Sheets, 14-16, 19, 20
Show, 12
ShowSettings, 20
Shp, 15, 18, 19
Sort, 13, 24
SortFields, 13, 24
SortOn, 13, 24
Split, 19
SplitLink, 18, 19
Start, 26
Subject, 26

T

Target, 3, 5
Text, 24
TextFrame2, 14, 15, 19, 24
TextRange, 14, 15, 19, 24
ThisWorkbook, 16, 19, 20
Top, 3, 15, 22, 24
TopLeftCell, 23

U

Ungroup, 20
UnGroupCal, 16, 20
Unique, 13, 24
Unload, 9, 13

V

Value, 11-17, 19-26
vbNo, 11, 12
vbYesNo, 11, 12
Visible, 3, 5, 10, 14-16, 18-20

W

Width, 24, 25
Worksheet, 15, 16
Worksheet_SelectionChange, 3, 5
WorksheetFunction, 11, 20, 21, 24
ws, 15, 16

X

xlAscending, 13, 24
xlFillDefault, 16, 17
xlFilterCopy, 13, 24
xlFormulas, 25
xlMove, 15, 18
xlSheetHidden, 16
xlSortNormal, 13, 24
xlSortOnValues, 13, 24
xlUp, 11-13, 24
xlWhole, 25

Y

Year, 14, 16, 25

Z

ZOrder, 15

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