

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 300,000 YouTube subscribers, 15,000,000 video views, 200+ comprehensive training videos, and a thriving 50,000 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



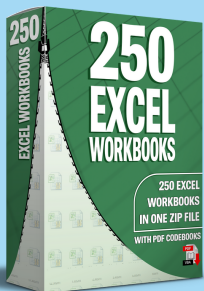
This comprehensive program will take you through a 12-phase process that will turn your enthusiasm for Excel into passive income.

[Click here to learn more](#)



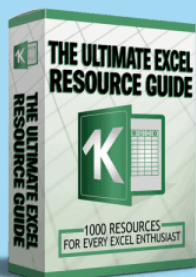
16 hour masterclass that will teach you the tips, tricks and techniques on how to create a dynamic single-click dashboard, and a ton more

[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](#)

Projects.....	3
VBAProject.....	3
Documents.....	3
Admin.....	3
(Declarations).....	3
Worksheet_Change [Sub].....	3
ApptDB.....	4
(Declarations).....	4
ApptItems.....	5
(Declarations).....	5
Worksheet_SelectionChange [Sub].....	5
Attachments.....	6
(Declarations).....	6
CalPopUp.....	7
(Declarations).....	7
Dashboard.....	8
(Declarations).....	8
InvItems.....	9
(Declarations).....	9
Invoice.....	10
(Declarations).....	10
Worksheet_Change [Sub].....	10
InvoiceList.....	12
(Declarations).....	12
Patients.....	13
(Declarations).....	13
PatMgr.....	14
(Declarations).....	14
Worksheet_Change [Sub].....	14
Worksheet_SelectionChange [Sub].....	14
Sheet3.....	15
(Declarations).....	15
Staff.....	16
(Declarations).....	16
ThisWorkbook.....	17
(Declarations).....	17
Modules.....	18
Application_Macros.....	18
(Declarations).....	18
ApptType_SetColor [Sub].....	18
BrowseForAttachFolder [Sub].....	18
BrowseForStaffPicFolder [Sub].....	18
Menu_GoToSheet [Sub].....	18
Appt_Macros.....	19
(Declarations).....	19
Appt_CreateLoadInvoice [Sub].....	19
Appt_Delete [Sub].....	19
Appt_Email [Sub].....	19
Appt_Load [Sub].....	20
Appt_New [Sub].....	20
Appt_SavedMsg [Sub].....	20
Appt_SaveUpdate [Sub].....	20
Attach_Macros.....	22
(Declarations).....	22
Attach_Add [Sub].....	22
Attach_Delete [Sub].....	22
Attach_Open [Sub].....	23
Attach_Preview [Sub].....	23
Invoice_Macros.....	25
(Declarations).....	25
Invoice_Delete [Sub].....	25
Invoice_Email [Sub].....	25
Invoice_Load [Sub].....	26
Invoice_New [Sub].....	26
Invoice_Print [Sub].....	26
Invoice_SavedMsg [Sub].....	26

Invoice_SaveUpdate [Sub]	27
Invoice_ViewAppt [Sub]	27
Patient_Macros	28
(Declarations)	28
Patient_BackToMainArea [Sub]	28
Patient_Delete [Sub]	28
Patient_Load [Sub]	28
Patient_LoadAppts [Sub]	28
Patient_LoadAttachments [Sub]	29
Patient_LoadInvoices [Sub]	29
Patient_New [Sub]	29
Patient_Save [Sub]	30
Patient_SavedMsg [Sub]	30
Patient_SortList [Sub]	30
Patient_ViewAllAppts [Sub]	31
Patient_ViewAllInvoices [Sub]	31
Patient_ViewInvoice [Sub]	31
Patient_ViewSelectedRecord [Sub]	31
PopUp_Calendar	32
(Declarations)	32
CalCol [Sub]	32
CalendarHide [Sub]	32
CalendarShow [Sub]	32
CalFormulaReplacement [Sub]	33
CheckForSheet [Sub]	33
CreateCalSht [Sub]	34
GroupCal [Sub]	36
MacroLinkRemover [Sub]	36
NextMonth [Sub]	37
NextYear [Sub]	37
PrevMonth [Sub]	37
PrevYear [Sub]	38
ReplaceCalendar [Sub]	38
SelectDay [Sub]	38
ShowSettings [Sub]	38
UnGroupCal [Sub]	39
Schedule_Macros	40
(Declarations)	40
Appt_CheckForMove [Sub]	40
Appt_CreateInvoice [Sub]	40
Appt_Select [Sub]	41
Sched_NextDay [Sub]	41
Sched_PrevDay [Sub]	41
Sched_Today [Sub]	41
Schedule_Refresh [Sub]	41
Schedule_ShowCalendar [Sub]	42

1 Option Explicit

2

3 Private Sub Worksheet_Change(ByVal Target As Range)

4 If Target.CountLarge > 1 Then Exit Sub

5 If Not Intersect(Target, Range("C18")) Is Nothing Then Patient_SortList

6 End Sub

7

1 Option Explicit

2

```
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     If Not Intersect(Target, Range("F3:F999" )) Is Nothing And Range("B" & Target.Row).
        Value <> Empty Then
7         With Shapes("ColorPalette" )
8             .Top = ActiveCell.Offset(1, 0).Top
9             .Left = ActiveCell.Left
10            .Visible = msoCTrue
11        End With
12    End If
13 End Sub
```


1 Option Explicit

2

1
2

Option Explicit

1 Option Explicit

2

1 Option Explicit

2

```

1 Option Explicit
2
3 Private Sub Worksheet_Change(ByVal Target As Range)
4     'On Patient Name Change
5     If Not Intersect(Target, Range("G5" )) Is Nothing And Range("G5" ).Value <> Empty
6         Then
7             If Target.Value <> Empty And Range("B2" ).Value <> Empty Then
8                 Dim PatRow As Long
9                 PatRow = Range("B2" ).Value 'Patient Row
10                Range("G6" ).Value = Patients.Range("H" & PatRow).Value 'Address 1
11                Range("G7" ).Value = Patients.Range("I" & PatRow).Value & ", " & Patients.
12                Range("J" & PatRow).Value & " " & Patients.Range("K" & PatRow).Value
13                'Address 2
14            End If
15        End If
16
17    'On Invoice Item Name Change (But not on Invoice Load)
18    If Not Intersect(Target, Range("F10:F31" )) Is Nothing And Range("B6" ).Value =
19    False Then
20        If Range("F" & Target.Row).Value <> Empty Then
21            Dim ItemRow As Long
22            On Error Resume Next
23            ItemRow = ApptItems.Range("ApptItem_Name" ).Find(Target.Value, , xlValues,
24            xlWhole).Row 'Item Row
25            On Error GoTo 0
26            If ItemRow <> 0 Then
27                Range("G" & Target.Row).Value = ApptItems.Range("D" & ItemRow).Value
28                'Item Description
29                Range("H" & Target.Row).Value = 1 'Set Default Qty
30                Range("I" & Target.Row).Value = ApptItems.Range("H" & ItemRow).Value
31                'Item Sales Price
32            End If
33        Else: 'Deleted Item (Clear Row)
34            Range("G" & Target.Row & ":I" & Target.Row).ClearContents
35            If Range("B" & Target.Row).Value <> "" Then
36                Dim InvItemDBRow As Long
37                InvItemDBRow = Range("B" & Target.Row).Value 'Invoice item DB Row
38                InvItems.Range(InvItemDBRow & ":" & InvItemDBRow).EntireRow.Delete
39                Range("B" & Target.Row).ClearContents
40            End If
41        End If
42    End If
43
44    'On Change of Invoice #, Load Invoice
45    If Not Intersect(Target, Range("E8" )) Is Nothing And Range("E8" ).Value <> Empty
46        Then
47            If Range("B4" ).Value = Empty Then
48                MsgBox "Invoice not found. Please enter another Invoice #"
49                Exit Sub
50            End If
51            Range("J3" ).Value = Range("E8" ).Value 'Copy over Searched Invoice ID
52            Invoice_Load 'Load Invoice
53        End If
54
55    'On change of Invoice Date or Terms but not on Invoice load
56    If Not Intersect(Target, Range("J4,J6" )) Is Nothing And Range("B6" ).Value = False
57        Then
58            If Range("J4" ).Value = Empty Or Range("J6" ).Value = Empty Then Exit Sub
59            Dim TermRow As Long
60            On Error Resume Next
61            TermRow = Admin.Range("Terms" ).Find(Range("J6" ).Value, , xlValues, xlWhole).Row

```

```
1 2
53 On Error GoTo 0
54 If TermRow <> 0 Then Range("J7" ).Value = Range("J4" ).Value + Admin.Range("F" &
    TermRow).Value 'Add in Term Days for Due Date
55 End If
56 End Sub
57
58
59
```

1 Option Explicit

2

1 Option Explicit

2


```

1 Option Explicit
2
3 Private Sub Worksheet_Change(ByVal Target As Range)
4     If Not Intersect(Target, Range("E4" )) Is Nothing And Range("E4" ).Value <> Empty
5         Then
6             If Range("B1" ).Value = Empty Then
7                 MsgBox "Please select a patient from the drop down list"
8                 Exit Sub
9             End If
10            Range("B2" ).Value = Range("B1" ).Value 'Bring Patient ID
11            Patient_Load
12        End If
13    End Sub
14
15 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
16     If Shapes("AttachGrp" ).Visible = True Then Shapes("AttachGrp" ).Visible = msoFalse
17     If Shapes("InvGrp" ).Visible = True Then Shapes("InvGrp" ).Visible = msoFalse
18     If Intersect(Target, Range("B5" )) Is Nothing Then
19         If Shapes("Calendar" ).Visible = True Then Shapes("Calendar" ).Visible = msoFalse
20     End If
21     If Target.CountLarge > 1 Then Exit Sub
22     'On Selection of Attachment Item
23     If Not Intersect(Target, Range("D21:D29" )) Is Nothing And Range("D" & Target.Row).
24     Value <> Empty Then
25         Range("B20" ).Value = Target.Row 'Set selected Row
26         With Shapes("AttachGrp" )
27             .Left = Range("D" & Target.Row).Left - 28
28             .Top = Range("D" & Target.Row).Top
29             .Visible = msoCTrue
30         End With
31         Attach_Preview 'Run Macro to Preview File
32     End If
33
34     'On Selection Of Previous Appointment
35     If Not Intersect(Target, Range("G11:J17" )) Is Nothing And Range("F" & Target.Row).
36     Value <> Empty Then
37         Range("B6" ).Value = Range("F" & Target.Row).Value
38         Appt_Load 'Run Macro To load appt
39     End If
40
41     'On Selection of Previous Invoiced
42     If Not Intersect(Target, Range("G21:J29" )) Is Nothing And Range("F" & Target.Row).
43     Value <> Empty Then
44         Range("B21" ).Value = Target.Row 'Set selected Row
45         With Shapes("InvGrp" )
46             .Left = Range("G" & Target.Row).Left - 14
47             .Top = Range("G" & Target.Row).Top
48             .Visible = msoCTrue
49         End With
50     End If
51
52     'On Selection of Show All Record
53     If Not Intersect(Target, Range("S4:W9999" )) Is Nothing And Range("S" & Target.Row)
54     .Value <> Empty Then Range("B22" ).Value = Target.Row
55 End Sub

```

1
2

Option Explicit

1 Option Explicit

2

1 Option Explicit

2

```
1 Option Explicit
2 Dim ShtName As String
3 Dim LastRow As Long, LastResultRow As Long
4
5 Sub ApptType_SetColor()
6     ActiveCell.Interior.Color = ApptItems.Shapes(Application.Caller).Fill.ForeColor.RGB
7     ApptItems.Shapes("ColorPalette").Visible = msoFalse
8 End Sub
9
10
11 Sub BrowseForAttachFolder()
12     Dim BillFolder As FileDialog
13     Set BillFolder = Application.FileDialog(msoFileDialogFolderPicker)
14     With BillFolder
15         .Title = "Browse For Patient Attachments Folder"
16         .AllowMultiSelect = False
17         If .Show <> -1 Then GoTo NoSelection
18         Admin.Range("C4").Value = .SelectedItems(1)
19     End With
20     NoSelection:
21 End Sub
22
23
24
25
26 Sub BrowseForStaffPicFolder()
27     Dim BillFolder As FileDialog
28     Set BillFolder = Application.FileDialog(msoFileDialogFolderPicker)
29     With BillFolder
30         .Title = "Browse For Staff Picture Folder"
31         .AllowMultiSelect = False
32         If .Show <> -1 Then GoTo NoSelection
33         Admin.Range("C3").Value = .SelectedItems(1)
34     End With
35     NoSelection:
36 End Sub
37
38 Sub Menu_GoToSheet()
39     Application.ScreenUpdating = False
40     ThisWorkbook.Sheets(Application.Caller).Activate
41     Application.ScreenUpdating = True
42 End Sub
```

```

1 Option Explicit
2 Dim ApptRow As Long, ApptCol As Long, LastRow As Long, LastResultRow As Long
3 Sub Appt_CreateLoadInvoice()
4     If PatMgr.Range("B7").Value = Empty Then
5         MsgBox "Please make sure to save the appointment before creating an Invoice"
6         Exit Sub
7     End If
8     Invoice.Activate
9     If PatMgr.Range("B9").Value = Empty Then 'Newly Created Invoice
10        Invoice_New
11        Invoice.Range("B5").Value = PatMgr.Range("B6").Value 'Set Appt ID
12        Invoice.Range("G5").Value = PatMgr.Range("E4").Value 'Set Patient Name
13        Invoice.Range("J4").Value = PatMgr.Range("H4").Value 'Set Invoice Date Same as
        Schedule Date
14        Invoice.Range("J5").Value = PatMgr.Range("H5").Value 'Set Assigned Staff
15        Invoice.Range("F10").Value = PatMgr.Range("J4").Value 'Set Appointment Item
16    Else 'Previously Created Invoice
17        Invoice.Range("E8").Value = PatMgr.Range("B9").Value 'Load Existing Invoice
18    End If
19 End Sub
20 Sub Appt_Delete()
21     If MsgBox("Are you sure you want to delete this appointment?" , vbYesNo, "Delete
        Appointment" ) = vbNo Then Exit Sub
22     With PatMgr
23         If .Range("B7").Value = Empty Then GoTo NotSaved
24         ApptRow = .Range("B7").Value 'Appointment Row
25         ApptDB.Range(ApptRow & ":" & ApptRow).EntireRow.Delete
26         NotSaved:
27         Appt_New
28         Patient_LoadAppts
29     End With
30 End Sub
31
32 Sub Appt_Email()
33     Dim OutApp As Object, OutMail As Object
34     Dim Subj As String, Mesg As String, PatEmail As String
35     Dim VarRow As Long
36     With PatMgr
37         If .Range("B7").Value = Empty Then 'Check For Unsaved Appointment
38             MsgBox "Please make sure to save the appointment before emailing"
39             Exit Sub
40         End If
41         PatEmail = .Range("E9").Value 'Patient Email
42         Set OutApp = CreateObject("Outlook.Application" )
43         Set OutMail = OutApp.CreateItem(0)
44         Subj = Admin.Range("P7").Value 'Subject
45         Mesg = Admin.Range("P8").Value 'Email Message
46         For VarRow = 8 To 18
47             Subj = Replace(Subj, Admin.Range("Q" & VarRow).Value, Format(Admin.Range("R"
                & VarRow).Value, Admin.Range("R" & VarRow).NumberFormat))
48             Mesg = Replace(Mesg, Admin.Range("Q" & VarRow).Value, Format(Admin.Range("R"
                & VarRow).Value, Admin.Range("R" & VarRow).NumberFormat))
49         Next VarRow
50         With OutMail
51             .To = PatEmail 'Patient email
52             .Subject = Subj 'Dynamic Subject
53             .body = Mesg 'Dynamic Message
54             .display 'To send use .Send
55         End With
56         On Error GoTo 0
57         Set OutMail = Nothing

```

1 2

```

1 2
58   End With
59   End Sub
60
61   Sub Appt_Load()
62       With PatMgr
63           .Range("B12" ).Value = True 'Clear any check for change loops
64           If .Range("B7" ).Value = Empty Then
65               MsgBox "Please select an appointment to view"
66               Exit Sub
67           End If
68           ApptRow = .Range("B7" ).Value 'Appointment Row
69           For ApptCol = 4 To 10
70               .Range(ApptDB.Cells(1, ApptCol).Value).Value = ApptDB.Cells(ApptRow, ApptCol).
71               Value 'Load Data
72           Next ApptCol
73           .Range("E4" ).Select
74       End With
75   End Sub
76
77   Sub Appt_New()
78       With PatMgr
79           .Range("B6,H4:H6,J4:J6,H7:J7" ).ClearContents
80           .Range("J4" ).Value = Admin.Range("C19" ).Value 'Default Appt. Item
81           .Range("H4" ).Select
82       End With
83   End Sub
84
85   Sub Appt_SavedMsg()
86       With PatMgr.Shapes("ApptSavedMsg" )
87           Dim i As Long, Delay As Double, StartTime As Double
88           .Visible = msoCTrue
89           For i = 1 To 100
90               .Fill.Transparency = i / 100
91               Delay = 0.009
92               StartTime = Timer
93               Do
94                   DoEvents
95               Loop While Timer - StartTime < Delay
96           Next i
97           .Visible = msoFalse
98       End With
99   End Sub
100
101   Sub Appt_SaveUpdate()
102       With PatMgr
103           'Check For required fields
104           If .Range("B10" ).Value < 3 Then
105               MsgBox "Please make sure to fill in a Date, Staff and Time before saving this
106               appointment"
107               Exit Sub
108           End If
109           If .Range("B7" ).Value = Empty Then 'New Appointment
110               ApptRow = ApptDB.Range("A99999" ).End(xlUp).Row + 1 'First Avail. Row
111               .Range("B6" ).Value = .Range("B8" ).Value 'Next Appt. ID
112               ApptDB.Range("A" & ApptRow).Value = .Range("B6" ).Value 'Next Appt. ID
113           Else 'Existing Appt.
114               ApptRow = .Range("B7" ).Value 'Existing Appt. Row
115           End If
116       End With

```

```
1 2
117 For ApptCol = 2 To 10
118     ApptDB.Cells(ApptRow, ApptCol).Value = .Range(ApptDB.Cells(1, ApptCol).Value).
        Value 'Save Data
119 Next ApptCol
120 .Range("B12").Value = True 'Set Appt Move to True to prevent loop
121 Patient_LoadAppts 'Run Macro To Load Appointments
122 Schedule_Refresh 'Refresh Schedule
123 Appt_SavedMsg 'Run Fade Out message
124 End With
125 End Sub
```



```

1 Option Explicit
2 Dim LastRow As Long, LastResultRow As Long, SelRow As Long, AttachRow As Long
3 Dim PatName As String, PatID As String, FilePath As String, FileName As String,
4 AttachFolder As String, FileType As String
5 Dim AttachFile As FileDialog
6
7 Sub Attach_Add()
8     With Attachments
9         If PatMgr.Range("B3").Value = Empty Then
10             MsgBox "Please save this patient before adding any attachments"
11             Exit Sub
12         End If
13         'Check For Attachment Folder
14         AttachFolder = [AttachmentFolder] 'Attachment Folder
15         If Dir(Admin.Range("AttachmentFolder").Value, vbDirectory) = "" Or Admin.Range(
16             "AttachmentFolder").Value = Empty Then
17             MsgBox "Please set a Attachment folder"
18             BrowseForAttachFolder 'Run macro to browse for Attach folder
19             If Dir(Admin.Range("AttachmentFolder").Value, vbDirectory) = "" Or Admin.
20                 Range("AttachmentFolder").Value = Empty Then Exit Sub
21             AttachFolder = Admin.Range("AttachmentFolder").Value 'Attachment Folder
22             (reset)
23         End If
24
25         PatID = PatMgr.Range("B2").Value 'Patient ID
26         PatName = PatMgr.Range("E4").Value 'Patient Name
27         AttachRow = .Range("A99999").End(xlUp).Row + 1 'First avail Row
28         Set AttachFile = Application.FileDialog(msoFileDialogFilePicker)
29         With AttachFile
30             .Title = "Browse for Attachment File"
31             .Filters.Add "Select File", "*.*", 1
32             .AllowMultiSelect = False
33             If .Show <> -1 Then GoTo NoSelection
34             FileName = Dir(.SelectedItems(1)) 'File Name
35             FilePath = AttachFolder & "\" & "Patient_" & PatID & "\" & Dir(FileName)
36             'Destination File Path
37             If FilePath = .SelectedItems(1) Then GoTo InCorrectFolder
38             If Dir(AttachFolder & "\" & "Patient_" & PatID, vbDirectory) = "" Then
39                 MkDir (AttachFolder & "\" & "Patient_" & PatID) 'Create Folder If needed
40                 for the Patient
41             If Dir(FilePath, vbDirectory) <> "" Then Kill (FilePath) 'Delete if exists
42             FileCopy .SelectedItems(1), FilePath 'Copy the file to the new path
43             InCorrectFolder:
44         End With
45         .Range("A" & AttachRow).Value = Now 'Set Added On Date & time
46         .Range("B" & AttachRow).Value = PatID 'Patient ID
47         .Range("C" & AttachRow).Value = FileName 'File Name
48         .Range("D" & AttachRow).Value = Right(FilePath, Len(FilePath) - InStrRev(
49             FilePath, ".")) 'File Type
50         .Range("E" & AttachRow).Value = "=Row()" 'Attachment Row
51         Patient_LoadAttachments 'Reload Attachments
52         PatMgr.Range("D21").Select 'Select First Row To Preview newly added attachment
53         Attach_Preview 'Preview Selected Attachment
54         NoSelection:
55     End With
56 End Sub
57
58 Sub Attach_Delete()
59     If MsgBox("Are you sure you want to delete this attachment?", vbYesNo, "Delete
60

```

1

```

1      Attachment" ) = vbNo Then Exit Sub
54     With PatMgr
55         On Error Resume Next
56         .Shapes("AttachPrev" ).Delete
57         On Error GoTo 0
58         If .Range("B20" ).Value = Empty Then
59             MsgBox "Please select an Attachment to delete"
60             Exit Sub
61         End If
62         SelRow = .Range("B20" ).Value
63         PatID = .Range("B2" ).Value 'Patient ID
64         AttachRow = .Range("C" & SelRow).Value 'Attach DB Row
65         If AttachRow = 0 Then GoTo NotSaved
66         On Error Resume Next
67         Kill ([AttachmentFolder] & "\" & "Patient_" & PatID & "\" & .Range("D" &
68             SelRow).Value) 'Delete Attachment File
69         Attachments.Range(AttachRow & ":" & AttachRow).EntireRow.Delete
70         NotSaved:
71         Patient_LoadAttachments 'Reload Attachments List
72         .Shapes("AttachGrp" ).Visible = msoFalse
73         .Range("B20" ).ClearContents 'Clear selected Row
74         .Range("E4" ).Select 'Select another cell
75     End With
76 End Sub
77 Sub Attach_Open()
78     With PatMgr
79         If .Range("B20" ).Value = Empty Then
80             MsgBox "Please select on a file to open"
81             Exit Sub
82         End If
83         SelRow = .Range("B20" ).Value 'Selected Row
84         If .Range("B2" ).Value = Empty Then
85             MsgBox "Please save the patient before opening up files"
86             Exit Sub
87         End If
88         PatID = .Range("B2" ).Value
89         'Check For Attachment Folder
90         AttachFolder = [AttachmentFolder] 'Attachment Folder
91         If Dir([AttachmentFolder], vbDirectory) = "" Or [AttachmentFolder] = Empty Then
92             MsgBox "Please set a Attachment folder"
93             BrowseForAttachFolder 'Run macro to browse for Attach folder
94             If Dir(Admin.Range("AttachmentFolder" ).Value, vbDirectory) = "" Or Admin.
95                 Range("AttachmentFolder" ).Value = Empty Then Exit Sub
96                 AttachFolder = Admin.Range("AttachmentFolder" ).Value 'Attachment Folder
97                 (reset)
98             End If
99             FilePath = AttachFolder & "\" & "Patient_" & PatID & "\" & .Range("D" &
100                 SelRow).Value 'Attachment File Path
101             ThisWorkbook.FollowHyperlink FilePath
102         End With
103 End Sub
104 Sub Attach_Preview()
105     With PatMgr
106         On Error Resume Next
107         .Shapes("AttachPrev" ).Delete
108         On Error GoTo 0
109         If .Range("B20" ).Value = Empty Then
110             MsgBox "Please select a Attachment to view"
111             Exit Sub

```

1 2 3

```

1 2 3
110 End If
111 'Check For Attachment Folder
112 AttachFolder = [AttachmentFolder] 'Attachment Folder
113 If Dir([AttachmentFolder], vbDirectory) = "" Or [AttachmentFolder] = Empty Then
114     MsgBox "Please set a Attachment folder"
115     BrowseForAttachFolder 'Run macro to browse for Attach folder
116     If Dir(Admin.Range("AttachmentFolder" ).Value, vbDirectory) = "" Or Admin.
Range("AttachmentFolder" ).Value = Empty Then Exit Sub
117     AttachFolder = Admin.Range("AttachmentFolder" ).Value 'Attachment Folder
(reset)
118 End If
119 SelRow = .Range("B20" ).Value 'Selected Attachment Row
120 If SelRow = 0 Then Exit Sub
121 AttachRow = .Range("C" & SelRow).Value 'Attachment DB Row
122 PatID = .Range("B2" ).Value 'Set Patient ID
123
124 FilePath = AttachFolder & "\" & "Patient_" & PatID & "\" & .Range("D" &
SelRow).Value 'Attachment File Path
125 If FilePath = Empty Then Exit Sub 'Exit on no Attachment
126 If Dir(FilePath, vbDirectory) = "" Then
127     MsgBox "Please make sure the Attachment path is correct, or browse for new
path"
128     Exit Sub
129 End If
130 FileType = Attachments.Range("D" & AttachRow).Value 'File Type
131 Select Case FileType
132     Case Is = "png" , "jpg" , "gif" , "jpeg" , "bmp" , "tiff" 'Picture Type
.Pictures.Insert(FilePath).Name = "AttachPrev"
133
134     Case Else
135         .OLEObjects.Add(FileName:=FilePath, Link:=False, DisplayAsIcon:=False).Name =
"AttachPrev"
136     End Select
137     With .Shapes("AttachPrev" )
138         .LockAspectRatio = True
139         If .Height > 90 Then .Height = 90 'Set max Height
140         If .Width > 90 Then .Width = 90 'Set Max Width
141         .Left = PatMgr.Range("E21" ).Left + ((PatMgr.Range("E:E" ).Width - .Width) / 2
) 'Set Left Centered Position
142         .Top = PatMgr.Range("E21" ).Top + ((PatMgr.Range("21:28" ).Height - .Height) /
2) 'Set Top Centered Position
143         If .Top < PatMgr.Range("E21" ).Top Then .Top = PatMgr.Range("E21" ).Top + 10
'Re-check top position
144     End With
145 End With
146 End Sub

```

```

1 Option Explicit
2 Dim InvRow As Long, InvCol As Long, InvItemRow As Long, LastRow As Long, LastItemRow As
  Long, LastInvoiceItemDBRow As Long, InvoiceItemDBRow As Long
3 Dim ResultRow As Long, LastResultRow As Long, ItemRow As Long
4 Dim FileName As String
5 Dim OutApp As Object, OutMail As Object
6
7 Sub Invoice_Delete()
8   If MsgBox("Are you sure you want to delete this Invoice?" , vbYesNo, "Delete
  Invoice" ) = vbNo Then Exit Sub
9   If Invoice.Range("B3" ).Value = "" Then GoTo NotSaved 'Invoice Not yet saved
10  InvRow = Invoice.Range("B3" ).Value 'Invoice Row
11  InvoiceList.Range(InvRow & ":" & InvRow).EntireRow.Delete 'Delete Invoice DB Row
12  With InvItems
13    LastInvoiceItemDBRow = .Range("A99999" ).End(xlUp).Row
14    .Range("A3:G" & LastInvoiceItemDBRow).AdvancedFilter xlFilterCopy, CriteriaRange
  :=.Range("I2:I3" ), CopyToRange:=.Range("K2:Q2" ), Unique:=True
15    LastResultRow = .Range("K99999" ).End(xlUp).Row 'Last Results Row
16    If LastResultRow < 3 Then GoTo NoDBItems
17    If LastResultRow < 4 Then GoTo NoSort
18    With .Sort
19      .SortFields.Clear
20      .SortFields.Add Key:=InvItems.Range("Q3" ), Order:=xlDescending 'Sort
  descending based on invocie DB row
21      .SetRange InvItems.Range("K3:Q" & LastResultRow)
22      .Apply
23    End With
24    NoSort:
25    For ResultRow = 3 To LastResultRow
26      InvoiceItemDBRow = .Range("Q" & ResultRow).Value 'Invoice Item Row
27      .Range(InvoiceItemDBRow & ":" & InvoiceItemDBRow).EntireRow.Delete
28    Next ResultRow
29    NoDBItems:
30    NotSaved:
31    Invoice_New
32    MsgBox "This Invoice has been deleted"
33  End With
34 End Sub
35
36
37 Sub Invoice_Email()
38   With Invoice
39     FileName = ThisWorkbook.Path & "\Invoice" & "_" & .Range("G5" ).Value & "_" &
  .Range("J3" ).Value & ".pdf" 'Set PDF Name to Patient & Inv. #
40     If Dir(FileName, vbDirectory) <> "" Then Kill (FileName) 'Delete file if exists
41
42     .ExportAsFixedFormat xlTypePDF, FileName, , , False
43     Set OutApp = CreateObject("Outlook.Application" )
44     Set OutMail = OutApp.createItem(0)
45
46     With OutMail
47       .To = Invoice.Range("B8" ).Value 'Patient Email
48       .Attachments.Add FileName
49       .Subject = "Please see the attached Invoice"
50       .body = "Hi " & Invoice.Range("G5" ).Value & vbCrLf & "Please see the
  attached Invoice #" & Invoice.Range("J3" ).Value & " for " & Format(Invoice.
  Range("J34" ).Value, "$#,##0.00" )
51       .display 'To send use .Send
52     End With
53   End With
  On Error GoTo 0
  Set OutMail = Nothing

```

```

54 | 1 2 End With
55 | End Sub
56 |
57 | Sub Invoice_Load()
58 |     With Invoice
59 |         If .Range("B3" ).Value = Empty Then
60 |             MsgBox "Please enter a correct Invoice #"
61 |             Exit Sub
62 |         End If
63 |
64 |         .Range("B6" ).Value = True 'Set Invoice Load To True
65 |         InvRow = .Range("B3" ).Value 'Invoice Row
66 |         .Range("B5,G5:G7,E9:E11,F10:I31,B10:B31" ).ClearContents 'Clear Invoice Fields
67 |         'Clear Invoice Fields
68 |
69 |         For InvCol = 2 To 11
70 |             If InvCol <> 4 Then .Range(InvoiceList.Cells(1, InvCol).Value).Value =
71 |                 InvoiceList.Cells(InvRow, InvCol).Value 'Load Invoice Data
72 |         Next InvCol
73 |
74 |         'Load Invoice Items
75 |         LastRow = InvItems.Range("A9999" ).End(xlUp).Row 'Last Item Row
76 |         If LastRow < 3 Then GoTo NoItems
77 |         InvItems.Range("A3:G" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=
78 |             InvItems.Range("I2:I3" ), CopyToRange:=InvItems.Range("K2:Q2" ), Unique:=False
79 |         LastResultRow = InvItems.Range("K9999" ).End(xlUp).Row
80 |         If LastResultRow < 3 Then GoTo NoItems
81 |         For ResultRow = 3 To LastResultRow
82 |             InvItemRow = InvItems.Range("P" & ResultRow).Value 'Invoice Row
83 |             .Range("F" & InvItemRow & ":I" & InvItemRow).Value = InvItems.Range("L" &
84 |                 ResultRow & ":O" & ResultRow).Value 'Item, Desc, Qty, Price
85 |             .Range("B" & InvItemRow).Value = InvItems.Range("Q" & ResultRow).Value
86 |             'Bring over item row
87 |         Next ResultRow
88 |         NoItems:
89 |         .Range("B6" ).Value = False 'Set Invoice Load To False
90 |     End With
91 | End Sub
92 |
93 | Sub Invoice_New()
94 |     With Invoice
95 |         .Range("B6" ).Value = True 'Invoice Load to True
96 |         .Range("B5,E8,J3,G5:G7,E9:E11,F10:I31,B10:B31" ).ClearContents 'Clear Invoice
97 |         Fields
98 |         .Range("J3" ).Value = Invoice.Range("B7" ).Value 'Set Next Invoice #
99 |         .Range("J4" ).Value = Date 'Set Current Date
100 |         .Range("B6" ).Value = False
101 |         .Range("G5" ).Select
102 |     End With
103 | End Sub
104 |
105 | Sub Invoice_Print()
106 |     Invoice.PrintOut , , , False, , , , False
107 | End Sub
108 |
109 | Sub Invoice_SavedMsg()
110 |     With Invoice.Shapes("InvSavedMsg" )
111 |         Dim i As Long, Delay As Double, StartTime As Double
112 |         .Visible = True
113 |         For i = 1 To 100
114 |             .Fill.Transparency = i / 100
115 |             Delay = 0.005
116 |             StartTime = Timer
117 |             Do

```

```

109 1 2 3 4 DoEvents
110     Loop While Timer - StartTime < Delay
111     Next i
112     .Visible = False
113 End With
114 End Sub

115
116 Sub Invoice_SaveUpdate()
117     With Invoice
118         'Determine New or Existing Invoice
119         If .Range("B3").Value = Empty Then 'New Invoice
120             InvRow = InvoiceList.Range("A9999").End(xlUp).Row + 1 'First Available Row
121
122             .Range("J3").Value = .Range("B7").Value 'Add Invoice Number
123             InvoiceList.Range("A" & InvRow).Value = .Range("B7").Value 'Add Invoice #
124             InvoiceList.Range("B" & InvRow).Value = .Range("B5").Value 'Appointment ID
125         Else: 'Existing Invoice
126             InvRow = .Range("B3").Value 'Existing Invoice Row
127         End If
128         For InvCol = 3 To 13
129             InvoiceList.Cells(InvRow, InvCol).Value = .Range(InvoiceList.Cells(1, InvCol).
130                 Value).Value 'Save/Update Data
131         Next InvCol
132
133         'Add / Update Invoice Items
134         LastItemRow = .Range("F31").End(xlUp).Row 'Last Invoice Item Row
135         If LastItemRow < 10 Then GoTo NoItems
136         For ItemRow = 10 To LastItemRow
137             If .Range("B" & ItemRow).Value <> Empty Then 'Existing Item Row
138                 InvItemRow = .Range("B" & ItemRow).Value 'Invoice Item DB Row
139             Else 'New Row
140                 InvItemRow = InvItems.Range("A9999").End(xlUp).Row + 1 'First Available
141                 Row
142                 InvItems.Range("A" & InvItemRow).Value = .Range("J3").Value 'Invoice #
143                 .Range("B" & ItemRow).Value = InvItemRow 'Invoice Item Row
144                 InvItems.Range("G" & InvItemRow).Value = "=Row()" 'Add Row #
145             End If
146             InvItems.Range("B" & InvItemRow & ":E" & InvItemRow).Value = .Range("F" &
147                 ItemRow & ":I" & ItemRow).Value 'Add Item, Desc, Qty, Price
148             InvItems.Range("F" & InvItemRow).Value = ItemRow 'Add/Update Invoice Row
149         Next ItemRow
150         NoItems:
151     End With
152     Invoice_SavedMsg 'Run Fade out message
153 End Sub

154 Sub Invoice_ViewAppt()
155     With Invoice
156         If .Range("B5").Value = Empty Or .Range("B2").Value = Empty Then 'On No
157             Patient or No Appointment
158             MsgBox "No Apppoint was created for this Invoice"
159             Exit Sub
160         End If
161         PatMgr.Range("E4").Value = .Range("G5").Value 'Set Patient Name
162         PatMgr.Range("B6").Value = .Range("B5").Value 'Set Appointment ID
163         PatMgr.Activate
164         Appt_Load
165     End With
166 End Sub

```

```

1 Option Explicit
2 Dim LastRow As Long, LastResultRow As Long, PatRow As Long, PatCol As Long, SelRow As
  Long
3 Dim PatName As String
4 Sub Patient_BackToMainArea()
5     ActiveWindow.ScrollColumn = 6
6     PatMgr.Range("B22").ClearContents
7 End Sub
8
9 Sub Patient_Delete()
10    If MsgBox("Are you sure you want to delete this Patient?" , vbYesNo, "Delete
  Patient" ) = vbNo Then Exit Sub
11    With PatMgr
12        If .Range("B3").Value = Empty Then GoTo NotSaved
13        PatRow = .Range("B3").Value 'Patient Row
14        Patients.Range(PatRow & ":" & PatRow).EntireRow.Delete
15        NotSaved:
16        Patient_New
17    End With
18 End Sub
19 Sub Patient_Load()
20    With PatMgr
21        .Range("B6,B20,B21,E5:E15,H4:H6,J4:J6,H7:J7,C21:D29,F11:J17,F21:J29").
  ClearContents
22        If .Range("B3").Value = Empty Then
23            MsgBox "Please select a correct patient to view"
24            Exit Sub
25        End If
26        PatRow = .Range("B3").Value 'Set Patient Row
27        For PatCol = 3 To 13
28            .Range("E" & PatCol + 2).Value = Patients.Cells(PatRow, PatCol).Value 'Load
  Patient Details
29        Next PatCol
30        .Range("D16").Value = Patients.Range("N" & PatRow).Value 'Add in Notes
31
32        Patient_LoadAttachments 'Load File Attachments
33        Patient_LoadAppts 'Load Appt. History
34        Patient_LoadInvoices 'Load Previous Invoices
35    End With
36 End Sub
37
38 Sub Patient_LoadAppts()
39    PatMgr.Range("F11:J17").ClearContents 'Clear Existing Appointments
40    With ApptDB
41        LastRow = .Range("A99999").End(xlUp).Row
42        If LastRow < 4 Then Exit Sub
43        .Range("A3:J" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
  "Q2:Q3" ), CopyToRange:=.Range("S2:W2" ), Unique:=True
44        LastResultRow = .Range("S99999").End(xlUp).Row
45        If LastResultRow < 3 Then Exit Sub
46        If LastRow < 4 Then GoTo SkipSort
47        'Sort Descending (Recently Added Appointments First)
48        With .Sort
49            .SortFields.Clear
50            .SortFields.Add Key:=ApptDB.Range("T3" ), Order:=xlDescending
51            .SetRange ApptDB.Range("S3:W" & LastResultRow)
52            .Apply
53        End With
54        SkipSort:
55        PatMgr.Range("F11:J17").Value = .Range("S3:W9").Value 'Bring Over Appt Data
56    End With

```

1


```

1
57 End Sub
58 Sub Patient_LoadAttachments()
59 PatMgr.Range("C21:D29").ClearContents 'Clear Existing Attachments
60 On Error Resume Next
61 PatMgr.Shapes("AttachPrev").Delete 'Clear Any Attachments
62 On Error GoTo 0
63 With Attachments
64 LastRow = .Range("A99999").End(xlUp).Row
65 If LastRow < 4 Then Exit Sub
66 .Range("A3:F" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
67 "H2:H3"), CopyToRange:=.Range("J2:K2"), Unique:=True
68 LastResultRow = .Range("J99999").End(xlUp).Row
69 If LastResultRow < 3 Then Exit Sub
70 If LastRow < 4 Then GoTo SkipSort
71 'Sort Desceding (Newest Attachments First)
72 With .Sort
73 .SortFields.Clear
74 .SortFields.Add Key:=Attachments.Range("J3"), Order:=xlDescending
75 .SetRange Attachments.Range("J3:K" & LastResultRow)
76 .Apply
77 End With
78 SkipSort:
79 PatMgr.Range("C21:D29").Value = .Range("J3:K11").Value 'Bring Over Attachment
80 Data
81 End With
82 End Sub
83
84 Sub Patient_LoadInvoices()
85 PatMgr.Range("F21:J29").ClearContents 'Clear Existing Invoices
86
87 With Invoicelist
88 LastRow = .Range("A99999").End(xlUp).Row
89 If LastRow < 4 Then Exit Sub
90 .Range("A3:M" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
91 "N2:N3"), CopyToRange:=.Range("Q2:U2"), Unique:=True
92 LastResultRow = .Range("Q99999").End(xlUp).Row
93 If LastResultRow < 3 Then Exit Sub
94 If LastRow < 4 Then GoTo SkipSort
95 'Sort Desceding (Recently Added Invoices First)
96 With .Sort
97 .SortFields.Clear
98 .SortFields.Add Key:=InvoiceList.Range("R3"), Order:=xlDescending
99 .SetRange InvoiceList.Range("Q3:U" & LastResultRow)
100 .Apply
101 End With
102 SkipSort:
103 PatMgr.Range("F21:J29").Value = .Range("Q3:U11").Value 'Bring Over Invoice
104 Data
105 End With
106 End Sub
107
108 Sub Patient_New()
109 With PatMgr
110 On Error Resume Next
111 .Shapes("AttachPrev").Delete
112 On Error GoTo 0
113 .Range("B2,B6,B20,B21,E4:E15,H4:H6,J4:J6,H7:J7,C21:D29,F11:J17,F21:J29").
114 ClearContents
115 .Range("D16").Value = "Notes" 'Set Default Text
116 .Range("E7").Value = Admin.Range("C19").Value 'Set Default Patient Type
117 .Range("E5").Select

```

1 2


```

1 2
113 End With
114
115 End Sub
116
117 Sub Patient_Save()
118 With PatMgr
119 If .Range("E5" ).Value = Empty Or .Range("E6" ).Value = Empty Then
120     MsgBox "Please make sure to add in a First Name and Last Name"
121     Exit Sub
122 End If
123 If .Range("B3" ).Value = Empty Then 'New Patient
124     PatRow = Patients.Range("B99999" ).End(xlUp).Row + 1 '1st Avail. Row
125     .Range("B2" ).Value = .Range("B4" ).Value 'New ID
126     Patients.Range("B" & PatRow).Value = .Range("B2" ).Value 'Set Patient ID
127 Else 'Existing Patient
128     PatRow = .Range("B3" ).Value 'Patient Row
129 End If
130 For PatCol = 3 To 13
131     Patients.Cells(PatRow, PatCol).Value = .Range("E" & PatCol + 2).Value
132     'Add/Update Patient Details
133 Next PatCol
134 If .Range("D16" ).Value <> "Notes" Then Patients.Range("N" & PatRow).Value = .
135     Range("D16" ).Value 'Add Notes if not default text
136 If [DisplayType] = "First Last" Then PatName = .Range("E6" ).Value & " " & .
137     Range("E5" ).Value Else PatName = .Range("E5" ).Value & ", " & .Range("E6" ).
138     Value 'Set Full Name
139 If .Range("E4" ).Value <> PatName Then .Range("E4" ).Value = PatName 'Set
140     Patient Name if applicable
141 End With
142 Patient_SortList 'Run Sort List
143 Patient_SavedMsg 'Run fade out message
144 End Sub
145 Sub Patient_SavedMsg()
146 With PatMgr.Shapes("PatSavedMsg" )
147     Dim i As Long, Delay As Double, StartTime As Double
148     .Visible = msoCTrue
149     For i = 1 To 100
150         .Fill.Transparency = i / 100
151         Delay = 0.009
152         StartTime = Timer
153         Do
154             DoEvents
155             Loop While Timer - StartTime < Delay
156         Next i
157     .Visible = msoFalse
158 End With
159 End Sub
160 Sub Patient_SortList()
161 With Patients
162     LastRow = .Range("B99999" ).End(xlUp).Row
163     If LastRow < 4 Then Exit Sub
164     .Range("AA3:AD9999" ).ClearContents 'Clear existing data
165     On Error Resume Next
166     .Names("Criteria" ).Delete 'Delete any criteria if applicable
167     On Error GoTo 0
168     .Range("B3:D" & LastRow).AdvancedFilter xlFilterCopy, , CopyToRange:=.Range(
169         "AA2:AC2" ), Unique:=True
170     LastResultRow = .Range("AA99999" ).End(xlUp).Row
171     If LastResultRow < 4 Then Exit Sub
172     .Range("AD3:AD" & LastResultRow).Formula = .Range("AD1" ).Formula 'Bring Down
173     Patient Name Formulas

```

1 2

```

1 2
167 With .Sort
168     .SortFields.Clear
169     .SortFields.Add Key:=Patients.Range("AD3" ), Order:=xlAscending
170     .SetRange Patients.Range("AA3:AD" & LastResultRow)
171     .Apply
172 End With
173 End With
174 End Sub
175
176 Sub Patient_ViewAllAppts()
177     PatMgr.Range("B22,S2:W9999" ).ClearContents
178     LastRow = ApptDB.Range("S99999" ).End(xlUp).Row
179     If LastRow < 3 Then
180         MsgBox "No appointments exist for this patient"
181         Exit Sub
182     End If
183     PatMgr.Range("S2" ).Value = "APPOINTMENT HISTORY"
184     PatMgr.Range("S3:W" & LastRow + 1).Value = ApptDB.Range("S2:W" & LastRow).Value
185     'Bring over Data
186     ActiveWindow.ScrollColumn = 18
187 End Sub
188
189 Sub Patient_ViewAllInvoices()
190     PatMgr.Range("B22,S2:W9999" ).ClearContents
191     LastRow = InvoiceList.Range("Q99999" ).End(xlUp).Row
192     If LastRow < 3 Then
193         MsgBox "No invoices exist for this patient"
194         Exit Sub
195     End If
196     PatMgr.Range("S2" ).Value = "INVOICE HISTORY"
197     PatMgr.Range("S3:W" & LastRow + 1).Value = InvoiceList.Range("Q2:U" & LastRow).
198     Value 'Bring over Data
199     ActiveWindow.ScrollColumn = 18
200 End Sub
201
202 Sub Patient_ViewInvoice()
203     If PatMgr.Range("B21" ).Value = Empty Then
204         MsgBox "Please select an Invoice to view"
205         Exit Sub
206     End If
207     PatMgr.Shapes("InvGrp" ).Visible = msoFalse
208     SelRow = PatMgr.Range("B21" ).Value 'Selected Row
209     Invoice.Range("E8" ).Value = PatMgr.Range("F" & SelRow).Value 'Set Invoice #
210     Invoice.Activate
211 End Sub
212
213 Sub Patient_ViewSelectedRecord()
214     With PatMgr
215         If .Range("B22" ).Value = Empty Then Exit Sub 'Exit on no selected row
216         SelRow = .Range("B22" ).Value 'Selected Row
217         If .Range("S2" ).Value = "INVOICE HISTORY" Then
218             Invoice.Range("E8" ).Value = PatMgr.Range("S" & SelRow).Value 'Set Invoice #
219             Invoice.Activate
220         Else ' Show Appointment
221             Range("B6" ).Value = PatMgr.Range("S" & SelRow).Value 'Set Appt ID
222             Appt_Load
223             Patient_BackToMainArea
224         End If
225     End With
226 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" ).Top = ActiveSheet.Range("A20" ).Top
20     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
21     ActiveSheet.ScreenUpdating = True
22     Sheets("CalPopUp" ).Range("A7" ).Value = ""
23     If Sheets("CalPopUp" ).Range("A20" ).Value <> Empty Then
24         For DayNum = 1 To 42
25             DayName = DayNum & "Day"
26             With ActiveSheet.Shapes(DayName)
27                 .Fill.ForeColor.RGB = RGB(255, 255, 255)
28                 .TextFrame2.TextRange.Font.Bold = msoFalse
29             End With
30         Next DayNum
31     End If
32     Exit Sub
33     NoCal: 'If calendar has been removed by accident, paste in backup calendar from
34         CalPopUp Sheet
35 End Sub
36 Sub CalendarShow()
37     With ActiveSheet
38         Set SelCell = Selection
39         'Check if active cell is a valid date
40         If IsDate(SelCell.Value) = True Then
41             Sheets("CalPopUp" ).Range("A1" ).Value = SelCell.Value
42         Else: 'If No Date or incorrect Date user current date
43             Sheets("CalPopUp" ).Range("A1" ).Value = "=Today()"
44         End If
45         'Clear all shapes to white (if calendar is visible)
46         If ActiveSheet.Shapes("Calendar" ).Visible = True Then
47             For DayNum = 1 To 42
48                 DayName = DayNum & "Day"
49                 With ActiveSheet.Shapes(DayName)
50                     .Fill.ForeColor.RGB = RGB(255, 255, 255)
51                     .TextFrame2.TextRange.Font.Bold = msoFalse
52                 End With
53             Next DayNum
54         End If
55
56         Sheets("CalPopUp" ).Range("A3" ).Value = Month(Sheets("CalPopUp" ).Range("A1" ).
57         Value) 'Set Month
58         Sheets("CalPopUp" ).Range("A2" ).Value = Year(Sheets("CalPopUp" ).Range("A1" ).
59         Value) 'Set Year
60         DayName = Sheets("CalPopUp" ).Range("A20" ).Value & "Day"

```

```

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

```

1

```

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

```

1 2

```

1 2
170 .Range("A8" ).AutoFill Destination:=.Range("A8:A19" ), Type:=xlFillDefault
171 .Range("A20" ).Value =
    "=IFERROR(INDIRECT(ADDRESS(SUMPRODUCT((B1:H6=A1)*ROW(B1:H6))+6,SUMPRODUCT((B1:H6=A
1)*COLUMN(B1:H6)),1,1))," & Chr(34) & Chr(34) & ")"
172 'Set Defined Names
173 ActiveWorkbook.Names.Add Name:="CalMonths" , RefersTo:="=CalPopUp!$A$8:$A$19"
174 ActiveWorkbook.Names.Add Name:="CalYear" , RefersTo:="=CalPopUp!$A$2"
175
176 'Add in Calendar Formulas
177
178 .Range("B1" ).Value =
    "=IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=1,DATE(CalYear,MATCH($A$4,Ca
lMonths,0),1)," & Chr(34) & Chr(34) & ")"
179 .Range("C1" ).Value = "=IF(B1<>" & Chr(34) & Chr(34) &
    ",B1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=2,DATE(CalYear,MATCH($
$4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")"
180 .Range("D1" ).Value = "=IF(C1<>" & Chr(34) & Chr(34) &
    ",C1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=3,DATE(CalYear,MATCH($
$4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")"
181 .Range("E1" ).Value = "=IF(D1<>" & Chr(34) & Chr(34) &
    ",D1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=4,DATE(CalYear,MATCH($
$4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")"
182 .Range("F1" ).Value = "=IF(E1<>" & Chr(34) & Chr(34) &
    ",E1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=5,DATE(CalYear,MATCH($
$4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")"
183 .Range("G1" ).Value = "=IF(F1<>" & Chr(34) & Chr(34) &
    ",F1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=6,DATE(CalYear,MATCH($
$4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")"
184 .Range("H1" ).Value = "=IF(G1<>" & Chr(34) & Chr(34) &
    ",G1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=7,DATE(CalYear,MATCH($
$4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")"
185 .Range("B2" ).Value = "=H1+1"
186 .Range("C2" ).Value = "=B2+1"
187 .Range("C2" ).AutoFill Destination:=.Range("C2:H2" ), Type:=xlFillDefault
188 .Range("B2:H2" ).AutoFill Destination:=.Range("B2:H4" ), Type:=xlFillDefault
189 .Range("B5" ).Value = "=IF(OR(H4=" & Chr(34) & Chr(34) & ",MONTH(H4+1)<>$A$3),"
    & Chr(34) & Chr(34) & ",H4+1)"
190 .Range("C5" ).Value = "=IFERROR(IF(MONTH(B5+1)<>$A$3," & Chr(34) & Chr(34) &
    ",B5+1)," & Chr(34) & Chr(34) & ")"
191 .Range("B6" ).Value = "=IFERROR(IF(OR(H5=" & Chr(34) & Chr(34) &
    ",MONTH(H5+1)<>$A$3)," & Chr(34) & Chr(34) & ",H5+1)," & Chr(34) & Chr(34) &
    ")"
192 .Range("C6" ).Value = "=IFERROR(F(MONTH(I5+1)<>$A$3," & Chr(34) & Chr(34) &
    ",I5+1)," & Chr(34) & Chr(34) & ")"
193 .Range("C5:C6" ).AutoFill Destination:=.Range("C5:H6" ), Type:=xlFillDefault
194
195 'Set format to Single Day
196 .Range("B1:H6" ).NumberFormat = "d"
197
198 'Add in relative Day #'s
199 .Range("B7" ).Value = "1"
200 .Range("C7" ).Value = "2"
201 .Range("B8" ).Value = "8"
202 .Range("C8" ).Value = "9"
203 .Range("B7:C8" ).AutoFill Destination:=.Range("B7:H8" ), Type:=xlFillDefault
204 .Range("B7:H8" ).AutoFill Destination:=.Range("B7:H12" ), Type:=xlFillDefault
205 GroupCal
206 End With
207 Exit Sub
208 NoCal:
209 MsgBox "The Pop-up Calendar does not exist on this worksheet. Please copy the

```

1

```

1      calendar over from another sheet and paste into this sheet"
210  End Sub
211  Sub GroupCal()
212  ActiveSheet.Shapes.Range(Array("NextTri" , "NextRec" )).Group.Select
213  Selection.ShapeRange.Name = "NextMonth"
214  ActiveSheet.Shapes.Range(Array("PrevTri" , "PrevRec" )).Group.Select
215  Selection.ShapeRange.Name = "PrevMonth"
216  ActiveSheet.Shapes.Range(Array("Settings" , "40Day" , "41Day" , "39Day" , "38Day" _
217      , "42Day" , "37Day" , "36Day" , "CalBack" , "Month" , "Year" , "CalBorder" ,
218      "1Day" , _
219      "3Day" , "14Day" , "7Day" , "4Day" , "2Day" , "5Day" , "8Day" , "10Day" ,
220      "6Day" , _
221      "13Day" , "11Day" , "9Day" , "12Day" , "15Day" , "17Day" , "20Day" , "21Day" ,
222      "18Day" _
223      , "16Day" , "19Day" , "22Day" , "24Day" , "26Day" , "27Day" , "25Day" , "23Day"
224      , _
225      "28Day" , "29Day" , "31Day" , "34Day" , "35Day" , "32Day" , "30Day" , "33Day" ,
226      "Sa" , _
227      "Fr" , "Th" , "We" , "Tu" , "Mo" , "Su" , "SetBtn" , "CalCol1" , "CalCol2" ,
228      "CalCol3" , _
229      "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" ,
230      "PrevMonth" , _
231      "NextMonth" , "NextYr" , "PrevYr" )).Visible = msoCTrue
232  ActiveSheet.Shapes.Range(Array("Settings" , "40Day" , "41Day" , "39Day" , "38Day" _
233      , "42Day" , "37Day" , "36Day" , "CalBack" , "Month" , "Year" , "CalBorder" , "1Day"
234      , _
235      "3Day" , "14Day" , "7Day" , "4Day" , "2Day" , "5Day" , "8Day" , "10Day" , "6Day" , _
236      "13Day" , "11Day" , "9Day" , "12Day" , "15Day" , "17Day" , "20Day" , "21Day" ,
237      "18Day"
238      , "16Day" , "19Day" , "22Day" , "24Day" , "26Day" , "27Day" , "25Day" , "23Day" , _
239      "28Day" , "29Day" , "31Day" , "34Day" , "35Day" , "32Day" , "30Day" , "33Day" ,
240      "Sa" , _
241      "Fr" , "Th" , "We" , "Tu" , "Mo" , "Su" , "SetBtn" , "CalCol1" , "CalCol2" ,
242      "CalCol3" , _
243      "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" , "PrevMonth"
244      , _
245      "NextMonth" , "NextYr" , "PrevYr" )).Select
246  Selection.ShapeRange.Group.Select
247  Selection.ShapeRange.Name = "Calendar"
248  Selection.Name = "Calendar"
249  Selection.Placement = xlMove
250  ActiveSheet.Shapes("Calendar" ).Placement = 2
251  End Sub
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 2
255 'Grab current macro link (if available)
256 On Error GoTo NextShp
257 MacroLink = Shp.OnAction
258
259 'Determine if shape was linking to a macro
260 If MacroLink <> "" And InStr(MacroLink, "!") <> 0 Then
261 'Split Macro Link at the exclamation mark (store in Array)
262 SplitLink = Split(MacroLink, "!")
263
264 'Pull text occurring after exclamation mark
265 NewLink = SplitLink(1)
266
267 'Remove any stragglng apostrophes from workbook name
268 If Right(NewLink, 1) = "'" Then
269 NewLink = Left(NewLink, Len(NewLink) - 1)
270 End If
271
272 'Apply New Link
273 Shp.OnAction = NewLink
274 End If
275 NextShp:
276 Next Shp
277 End Sub
278 Sub NextMonth()
279 'Next Month button
280 If Sheets("CalPopUp").Range("A20").Value <> Empty Then
281 DayName = Sheets("CalPopUp").Range("A20").Value & "Day"
282 With ActiveSheet.Shapes(DayName)
283 .Fill.ForeColor.RGB = RGB(255, 255, 255)
284 .TextFrame2.TextRange.Font.Bold = msoFalse
285 End With
286 End If
287 With Sheets("CalPopUp")
288 If .Range("A3").Value = 12 Then
289 .Range("A3").Value = 1
290 .Range("A2").Value = .Range("A2").Value + 1
291 Else:
292 .Range("A3").Value = .Range("A3").Value + 1
293 End If
294 If .Range("A6").Value > 0 Then
295 ActiveSheet.Shapes.Range(Array("36Day", "37Day", "38Day", "39Day",
296 "40Day", "41Day", "42Day")).Visible = True
297 Else:
298 ActiveSheet.Shapes.Range(Array("36Day", "37Day", "38Day", "39Day",
299 "40Day", "41Day", "42Day")).Visible = False
300 End If
301 End With
302 End Sub
303 Sub NextYear()
304 ThisWorkbook.Sheets("CalPopUp").Range("A2").Value = ThisWorkbook.Sheets(
305 "CalPopUp").Range("A2").Value + 1
306 End Sub
307 Sub PrevMonth()
308 'Previous Month Button
309 If Sheets("CalPopUp").Range("A20").Value <> Empty Then
310 DayName = Sheets("CalPopUp").Range("A20").Value & "Day"
311 With ActiveSheet.Shapes(DayName)
312 .Fill.ForeColor.RGB = RGB(255, 255, 255)
313 .TextFrame2.TextRange.Font.Bold = msoFalse
314 End With

```



```

1 2
313 End If
314 With Sheets("CalPopUp" )
315     If .Range("A3" ).Value = 1 Then
316         .Range("A3" ).Value = 12
317         .Range("A2" ).Value = .Range("A2" ).Value - 1
318     Else:
319         .Range("A3" ).Value = .Range("A3" ).Value - 1
320     End If
321     If .Range("A6" ).Value > 0 Then
322         ActiveSheet.Shapes.Range(Array("36Day" , "37Day" , "38Day" , "39Day" ,
323             "40Day" , "41Day" , "42Day" )).Visible = True
324     Else:
325         ActiveSheet.Shapes.Range(Array("36Day" , "37Day" , "38Day" , "39Day" ,
326             "40Day" , "41Day" , "42Day" )).Visible = False
327     End If
328 End With
329 End Sub
330
331 Sub PrevYear()
332     ThisWorkbook.Sheets("CalPopUp" ).Range("A2" ).Value = ThisWorkbook.Sheets(
333         "CalPopUp" ).Range("A2" ).Value - 1
334 End Sub
335
336 Sub ReplaceCalendar() 'Shape Deleted
337     CalPopUp.Shapes("Calendar" ).Copy 'Copy From Developers sheet
338     ActiveCell.Select 'Select the active Cell
339     ActiveSheet.Paste 'Paste in Calendar
340 End Sub
341
342 .....
343 ''''Select Day Of The Month
344 .....
345
346 Sub SelectDay()
347     Dim DayNumb As Long, RowNumb As Long, ColNumb As Long
348     DayNumb = Replace(Application.Caller, "Day" , "" )
349     RowNumb = Application.WorksheetFunction.RoundUp(DayNumb / 7, 0)
350     ColNumb = DayNumb Mod 7 + 1
351     If DayNumb Mod 7 = 0 Then
352         ColNumb = 8
353     Else
354         ColNumb = DayNumb Mod 7 + 1
355     End If
356
357     'On Error Resume Next
358     If ThisWorkbook.Sheets("CalPopUp" ).Range("A7" ).Value = Empty Then Exit Sub
359     ActiveSheet.Range(ThisWorkbook.Sheets("CalPopUp" ).Range("A7" ).Value).Value =
360     ThisWorkbook.Sheets("CalPopUp" ).Cells(RowNumb, ColNumb).Value
361     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
362     If ActiveCell.Address = "$B$5" Then Schedule_Refresh 'Refresh on Date Change
363     ActiveCell.Offset(0, 1).Select
364 End Sub
365
366 Sub ShowSettings()
367     'Show or Hide Calendar Settings Panel
368     If ActiveSheet.Shapes.Range(Array("Settings" )).Visible = True Then
369         ActiveSheet.Shapes.Range(Array("Settings" , "CalCol1" , "CalCol2" , "CalCol3" ,
370             "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" )).Visible
371         = False
372     Else:
373         ActiveSheet.Shapes.Range(Array("Settings" , "CalCol1" , "CalCol2" , "CalCol3" ,
374             "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" )).Visible
375         = True
376     End If
377 End Sub

```

```
366     1 End Sub
367
368     Sub UnGroupCal()
369         On Error Resume Next
370         ActiveSheet.Shapes("Calendar").Ungroup
371         ActiveSheet.Shapes("NextMonth").Ungroup
372         ActiveSheet.Shapes("PrevMonth").Ungroup
373         On Error GoTo 0
374     End Sub
```

```

1 Option Explicit
2 Dim ScRow As Long, ScCol As Long, ScRows As Long, CountDelay As Long, ApptRow As Long
3 Dim LastRow As Long, LastResultRow As Long, ResultRow As Long, ItemRow As Long
4 Dim StaffNm As String, ApptID As String, ApptCol As String, ApptItem As String, PatNm
  As String
5 Dim Duration As Double, ScInterval As Double, ScStartTime As Double
6 Dim ApptDt As Date, ApptStart As Date, ApptEnd As Date
7 Dim ApptShp As Shape

```

```

10 Sub Appt_CheckForMove()
11     With PatMgr
12         If .Range("B6").Value = Empty Then Exit Sub
13         ApptID = .Range("B6").Value 'Appt ID
14
15         For CountDelay = 1 To 100000
16             DoEvents
17             If .Range("B12").Value = True Then End 'Exit Loop on Move Appt Move = True
18
19             With .Shapes("ApptItem" & ApptID)
20                 If .Left <> PatMgr.Range("B13").Value Or .Top <> PatMgr.Range("B14").
  Value Then 'Move Detect
21                     'Check for incorrect move or missing Supervisor
22                     If .Left < PatMgr.Range("L1").Left Or PatMgr.Cells(3, .TopLeftCell.
  Column).Value = 0 Or .Top < PatMgr.Range("A3").Top - 1 Or .Left >
  PatMgr.Range("R1").Left Or PatMgr.Range("L" & .TopLeftCell.Row).Value
  = 0 Then
23                         MsgBox "Plase make sure to move the Appointment to a correct time
  and staff on the schedule"
24                         Schedule_Refresh
25                         Exit Sub
26                     End If
27                     StaffNm = PatMgr.Cells(3, .TopLeftCell.Column).Value 'Staff Name
28                     PatMgr.Range("H5").Value = StaffNm 'Set Staff Name
29                     ApptStart = PatMgr.Range("L" & .TopLeftCell.Row).Value 'Set New Start
  Time
30                     PatMgr.Range("J5").Value = ApptStart 'Update Start Time
31                     PatMgr.Range("B12").Value = True 'Set Move Appointment to True
32                     Appt_SaveUpdate 'Save Appt Changes
33                 End If
34             End With
35         Next CountDelay
36         .Range("B12").Value = True 'Set Move Apptment to True
37     End With
38 End Sub

```

```

40 Sub Appt_CreateInvoice()
41     With PatMgr
42         .Range("B9").Value = True 'Set Move Appointment to True to exit any loops
43         If .Range("B3").Value = Empty Then
44             MsgBox "Please select an appointment to create an invoice"
45             Exit Sub
46         End If
47         Invoice_New 'Clear existing Invoice
48         If .Range("B12").Value <> Empty Then 'Existing Invoice
49             Invoice.Range("J1").Value = InvoiceList.Range("A" & .Range("B12").Value).
  Value 'Set current invoice # to search
50         Else ' New Invoice
51             ApptRow = .Range("B3").Value 'Appointment Row
52             Invoice.Range("E4").Value = ApptDB.Range("B" & ApptRow).Value 'Client Name
53             Invoice.Range("H2").Value = .Range("D5").Value 'Set Invoice Date Default to

```

1 2 3

```

1 2 3
Appt. Date
54 Invoice.Range("G7" ).Value = ApptDB.Range("G" & ApptRow).Value 'Staff
55 Invoice.Range("D9" ).Value = ApptDB.Range("I" & ApptRow).Value 'Primary
Service
56 Invoice.Range("D10" ).Value = ApptDB.Range("J" & ApptRow).Value 'Secondary
Service
57 Invoice.Range("B5" ).Value = .Range("B2" ).Value 'Appointment ID
58 Invoice.Range("H1" ).Value = Invoice.Range("B7" ).Value 'Next Invoice ID
59 End If
60 Invoice.Activate
61 End With
62 End Sub
63
64
65
66 Sub Appt_Select()
67 With PatMgr
68 ApptID = Replace(Application.Caller, Left(Application.Caller, 8), "" )
69 .Range("B6" ).Value = ApptID
70 ApptRow = .Range("B7" ).Value 'Appointent Row
71 .Range("E4" ).Value = ApptDB.Range("C" & ApptRow).Value 'Patient Name
72 .Range("B6" ).Value = ApptID
73 Appt_Load 'Load Appt
74 'On Error GoTo NoAppt
75 Set ApptShp = .Shapes(Application.Caller)
76 .Range("B12" ).Value = False 'Set Appt Change to false
77 .Range("B13" ).Value = ApptShp.Left 'Set initial left position
78 .Range("B14" ).Value = ApptShp.Top ' Set Initial top position
79 .Shapes(Application.Caller).Select
80 .Range("B12" ).Value = False 'Set Appt Move to False
81 Appt_CheckForMove 'Run Macro To check for move
82 NoAppt:
83 End With
84 End Sub
85 Sub Sched_NextDay()
86 PatMgr.Range("B5" ).Value = PatMgr.Range("B5" ).Value + 1
87 Schedule_Refresh
88 End Sub
89 Sub Sched_PrevDay()
90 PatMgr.Range("B5" ).Value = PatMgr.Range("B5" ).Value - 1
91 Schedule_Refresh
92 End Sub
93
94 Sub Sched_Today()
95 PatMgr.Range("B5" ).Value = Date 'Current
96 Schedule_Refresh
97 End Sub
98
99 Sub Schedule_Refresh()
100 For Each ApptShp In PatMgr.Shapes
101 If InStr(ApptShp.Name, "ApptItem" ) > 0 Then ApptShp.Delete 'Delete any existing
appointments
102 Next ApptShp
103 ScStartTime = Admin.Range("StartTime" ).Value 'Start time
104 ScInterval = Admin.Range("Interval" ).Value 'Schedual Interval
105 With ApptDB
106 .Range("AA3:AI999" ).ClearContents 'Clear prior results
107 LastRow = .Range("A99999" ).End(xlUp).Row
108 If LastRow < 4 Then Exit Sub
109 .Range("A3:J" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
"02:03" ), CopyToRange:=.Range("AA2:AI2" ), Unique:=True

```

```

1 2
110 LastResultRow = .Range("AA99999" ).End(xlUp).Row
111 If LastResultRow < 3 Then Exit Sub
112 For ResultRow = 3 To LastResultRow
113     ApptID = .Range("AA" & ResultRow).Value 'Appt. ID
114     PatNm = .Range("AC" & ResultRow).Value 'Patient Name
115     ApptStart = .Range("AD" & ResultRow).Value 'Start Time
116     Duration = .Range("AE" & ResultRow).Value 'Duration
117     StaffNm = .Range("AH" & ResultRow).Value 'Staff Name
118     ScRows = (Duration / ScInterval) 'Pat Mgr Rows
119     ApptItem = .Range("AI" & ResultRow).Value 'Appointment Item
120     On Error Resume Next
121     ItemRow = ApptItems.Range("ApptItem_Name" ).Find(ApptItem, , xlValues, xlWhole
122     ).Row
123     On Error GoTo 0
124     If ItemRow <> 0 Then ApptCol = ApptItems.Range("F" & ItemRow).Interior.Color
125     Else ApptCol = "" 'Get Appt. Item Color
126
127     ScRow = (ApptStart - ScStartTime) / ScInterval + 4 'PatMgr Row
128     On Error Resume Next
129     ScCol = Admin.Range("SchedStaff" ).Find(StaffNm, , xlValues, xlWhole).Row + 2
130     'Set PatMgr column
131     On Error GoTo 0
132     If ScCol = 0 Then GoTo NextAppt
133     'Create Appointment Shape
134     PatMgr.Shapes("SampleAppt" ).Duplicate.Name = "ApptItem" & ApptID
135     With PatMgr.Shapes("ApptItem" & ApptID)
136         If ApptCol <> "" Then .Fill.ForeColor.RGB = ApptCol 'Set Back color if
137         available
138         .TextFrame2.TextRange.Text = PatNm & vbCrLf & ApptItem 'Set Patient Name
139         & Appt Item
140         .Left = PatMgr.Cells(ScRow, ScCol).Left
141         .Top = PatMgr.Cells(ScRow, ScCol).Top
142         .Height = PatMgr.Cells(ScRow, ScCol).Height * ScRows 'Shape Height
143         .Width = PatMgr.Cells(ScRow, ScCol).Width 'Set Width
144     End With
145     NextAppt:
146     Next ResultRow
147 End With
148 End Sub
149
150 Sub Schedule_ShowCalendar()
151     Application.ScreenUpdating = False
152     With PatMgr.Shapes("Calendar" )
153         .Top = PatMgr.Range("P4" ).Top
154         .Left = PatMgr.Range("P4" ).Left - 15
155         .ZOrder msoBringToFront
156         .Visible = msoCTrue
157     End With
158     PatMgr.Range("B5" ).Select
159     CalPopUp.Range("A7" ).Value = "$B$5" 'Set Scheduled Date Cell
160     ActiveWindow.ScrollColumn = 3
161     Application.ScreenUpdating = True
162 End Sub

```

—
_, 36

A

Activate, 18, 19, 27, 31, 34, 41
 ActiveCell, 5, 18, 33, 38
 ActiveSheet, 32-34, 36-39
 ActiveWindow, 28, 31, 42
 ActiveWorkbook, 33-35
 ActSht, 34
 Add, 22, 24, 25, 28, 29, 31, 34, 35
 Address, 33, 34, 38
 Admin, 10, 11, 18-20, 22-24, 29, 41, 42
 AdvancedFilter, 25, 26, 28-30, 41
 After, 34
 AllowMultiSelect, 18, 22
 Application, 18, 22, 32, 38, 41, 42
 Apply, 25, 28, 29, 31
 Appt_CheckForMove, 40, 41
 Appt_CreateInvoice, 40
 Appt_CreateLoadInvoice, 19
 Appt_Delete, 19
 Appt_Email, 19
 Appt_Load, 14, 20, 27, 31, 41
 Appt_New, 19, 20
 Appt_SavedMsg, 20, 21
 Appt_SaveUpdate, 20, 40
 Appt_Select, 41
 ApptCol, 19-21, 40, 42
 ApptDB, 19-21, 28, 31, 40, 41
 ApptDt, 40
 ApptEnd, 40
 ApptID, 40-42
 ApptItem, 40, 42
 ApptItems, 10, 18, 42
 ApptRow, 19-21, 40, 41
 ApptShp, 40, 41
 ApptStart, 40, 42
 ApptType_SetColor, 18
 Array, 32, 33, 36-38
 Attach_Add, 22
 Attach_Delete, 22
 Attach_Open, 23
 Attach_Preview, 14, 22, 23
 AttachFile, 22
 AttachFolder, 22-24
 AttachmentFolder, 22-24
 Attachments, 22-25, 29
 AttachRow, 22-24
 AutoFill, 35

B

BillFolder, 18
 body, 19, 25
 Bold, 32, 33, 37
 BrowseForAttachFolder, 18, 22-24
 BrowseForStaffPicFolder, 18

C

CalCol, 32, 34
 CalendarHide, 32
 CalendarShow, 32
 CalFormulaReplacement, 33
 Caller, 18, 32, 38, 41
 CalPopUp, 38, 42
 Cells, 20, 21, 26-28, 30, 33, 34, 38, 40, 42
 CheckForSheet, 33
 Chr, 34, 35
 Clear, 25, 28, 29, 31

ClearContents, 10, 20, 23, 26, 28-31, 41
 ColCnt, 34
 ColNum, 33
 ColNumb, 38
 Color, 18, 42
 Column, 40
 Copy, 34, 38
 CopyToRange, 25, 26, 28-30, 41
 Count, 34
 CountDelay, 40
 CountLarge, 3, 5, 14
 CreateCalSht, 34
 createItem, 19, 25
 CreateObject, 19, 25
 CreateWS, 33, 34
 CriteriaRange, 25, 26, 28, 29, 41

D

DayCnt, 34
 DayName, 32, 33, 37
 DayNum, 32, 33
 DayNumb, 38
 Delay, 20, 26, 27, 30
 Delete, 10, 19, 23, 25, 28-30, 41
 Destination, 35
 Dir, 22-25
 display, 19, 25
 DisplayAsIcon, 24
 DisplayType, 30
 DoEvents, 20, 27, 30, 40
 DrawingObject, 33, 34
 Duplicate, 42
 Duration, 40, 42

E

Empty, 5, 10, 14, 19, 20, 22-24, 26-28, 30-32, 37, 38, 40
 EntireRow, 10, 19, 23, 25, 28
 Explicit, 3-10, 12-19, 22, 25, 28, 40
 ExportAsFixedFormat, 25

F

FileCopy, 22
 FileDialog, 18, 22
 FileName, 22, 24, 25
 FilePath, 22-24
 FileType, 22, 24
 Fill, 18, 20, 26, 30, 32, 33, 37, 42
 Filters, 22
 Find, 10, 42
 FollowHyperlink, 23
 Font, 32, 33, 37
 ForeColor, 18, 32, 33, 37, 42
 Format, 19, 25
 Formula, 30, 33, 34

G

Group, 36
 GroupCal, 35, 36
 GroupItems, 33

H

Height, 24, 42

I

i, 20, 26, 27, 30
 InCorrectFolder, 22
 Insert, 24
 InStr, 33, 37, 41
 InStrRev, 22

Interior, 18, 42
 Intersect, 3, 5, 10, 14
 InvCol, 25-27
 InvItemDBRow, 10
 InvItemRow, 25-27
 InvItems, 10, 25-27
 Invoice, 19, 25-27, 31, 40, 41
 Invoice_Delete, 25
 Invoice_Email, 25
 Invoice_Load, 10, 26
 Invoice_New, 19, 25, 26, 40
 Invoice_Print, 26
 Invoice_SavedMsg, 26, 27
 Invoice_SaveUpdate, 27
 Invoice_ViewAppt, 27
 InvoiceItemDBRow, 25
 InvoiceList, 25-27, 29, 31, 40
 InvRow, 25-27
 IsDate, 32
 ItemRow, 10, 25, 27, 40, 42

K

Key, 25, 28, 29, 31
 Kill, 22, 23, 25

L

LastInvoiceItemDBRow, 25
 LastItemRow, 25, 27
 LastResultRow, 18, 19, 22, 25, 26, 28-31, 40, 42
 LastRow, 18, 19, 22, 25, 26, 28-31, 40, 41
 Left, 5, 14, 24, 33, 37, 40-42
 Len, 22, 37
 Link, 24
 LockAspectRatio, 24

M

MacroLink, 36, 37
 MacroLinkRemover, 33, 36
 Menu_GoToSheet, 18
 Mesg, 19
 Mkdir, 22
 Month, 32, 34
 MsgBox, 10, 14, 19, 20, 22-28, 30, 31, 33, 35, 40
 msoBringToFront, 33, 42
 msoCTrue, 5, 14, 20, 30, 33, 36, 42
 msoFalse, 5, 14, 18, 20, 23, 30-32, 37, 38
 msoFileDialogFilePicker, 22
 msoFileDialogFolderPicker, 18
 msoTrue, 33

N

Name, 24, 34-36, 41, 42
 Names, 30, 35
 NewLink, 36, 37
 NextAppt, 42
 NextMonth, 37
 NextShp, 37
 NextYear, 37
 NoAppt, 41
 NoCal, 32, 33, 35
 NoDBItems, 25
 Noltems, 26, 27
 NoSelection, 18, 22
 NoSort, 25
 NotSaved, 19, 23, 25, 28
 Now, 22
 NumberFormat, 19, 35

O

Offset, 5, 33, 38
 OLEObjects, 24
 OnAction, 33, 34, 37
 Order, 25, 28, 29, 31
 OutApp, 19, 25
 OutMail, 19, 25

P

Paste, 34, 38
 PatCol, 28, 30
 PatEmail, 19
 Path, 25
 PatID, 22-24
 Patient_BackToMainArea, 28, 31
 Patient_Delete, 28
 Patient_Load, 14, 28
 Patient_LoadAppts, 19, 21, 28
 Patient_LoadAttachments, 22, 23, 28, 29
 Patient_LoadInvoices, 28, 29
 Patient_New, 28, 29
 Patient_Save, 30
 Patient_SavedMsg, 30
 Patient_SortList, 3, 30
 Patient_ViewAllAppts, 31
 Patient_ViewAllInvoices, 31
 Patient_ViewInvoice, 31
 Patient_ViewSelectedRecord, 31
 Patients, 10, 28, 30, 31
 PatMgr, 19, 20, 22-24, 27-31, 40-42
 PatName, 22, 28, 30
 PatNm, 40, 42
 PatRow, 10, 28, 30
 Pictures, 24
 Placement, 33, 36
 PrevMonth, 37
 PrevYear, 38
 PrintOut, 26

R

Range, 3, 5, 10, 11, 14, 18-38, 40-42
 RefersTo, 35
 Replace, 19, 38, 41
 ReplaceCalendar, 38
 ResultRow, 25, 26, 40, 42
 RGB, 18, 32, 33, 37, 42
 Right, 22, 37
 RoundUp, 38
 Row, 5, 10, 14, 20, 22, 25-31, 40-42
 RowCnt, 34
 RowNum, 33
 RowNumb, 38

S

ScCol, 40, 42
 Sched_NextDay, 41
 Sched_PrevDay, 41
 Sched_Today, 41
 Schedule_Refresh, 21, 38, 40, 41
 Schedule_ShowCalendar, 42
 ScInterval, 40-42
 ScreenUpdating, 18, 32, 42
 ScrollColumn, 28, 31, 42
 ScRow, 40, 42
 ScRows, 40, 42
 ScStartTime, 40-42
 SelCell, 32, 33
 SelectDay, 38
 SelectedItems, 18, 22

Selection, 32, 36
 SelRow, 22-24, 28, 31
 SetRange, 25, 28, 29, 31
 Shape, 33, 36, 40
 ShapeRange, 32, 36
 Shapes, 5, 14, 18, 20, 23, 24, 26, 29-34, 36-42
 Sheets, 18, 32-34, 37, 38
 Show, 18, 22
 ShowSettings, 38
 Shp, 33, 36, 37
 ShtName, 18
 SkipSort, 28, 29
 Sort, 25, 28, 29, 31
 SortFields, 25, 28, 29, 31
 Split, 37
 SplitLink, 36, 37
 StaffNm, 40, 42
 StartTime, 20, 26, 27, 30
 Subj, 19
 Subject, 19, 25

T

Target, 3, 5, 10, 14
 TermRow, 10, 11
 Text, 42
 TextFrame2, 32, 33, 37, 42
 TextRange, 32, 33, 37, 42
 ThisWorkbook, 18, 23, 25, 34, 37, 38
 Timer, 20, 26, 27, 30
 Title, 18, 22
 Top, 5, 14, 24, 32, 33, 40-42
 TopLeftCell, 40
 Transparency, 20, 26, 30

U

Ungroup, 39
 UnGroupCal, 34, 39
 Unique, 25, 26, 28-30, 41

V

Value, 5, 10, 11, 14, 18-35, 37, 38, 40-42
 VarRow, 19
 vbCrLf, 25, 42
 vbDirectory, 22-25
 vbNo, 19, 23, 25, 28
 vbYesNo, 19, 22, 25, 28
 Visible, 5, 14, 18, 20, 23, 26, 27, 30-34, 36-38, 42

W

Width, 24, 42
 Worksheet, 33, 34
 Worksheet_Change, 3, 10, 14
 Worksheet_SelectionChange, 5, 14
 WorksheetFunction, 38
 ws, 33, 34

X

xlAscending, 31
 xlDescending, 25, 28, 29
 xlFillDefault, 35
 xlFilterCopy, 25, 26, 28-30, 41
 xlMove, 33, 36
 xlSheetHidden, 34
 xlTypePDF, 25
 xlUp, 20, 22, 25-31, 41, 42
 xlValues, 10, 42
 xlWhole, 10, 42

Y

Year, 32, 34

Z

ZOrder, 33, 42

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