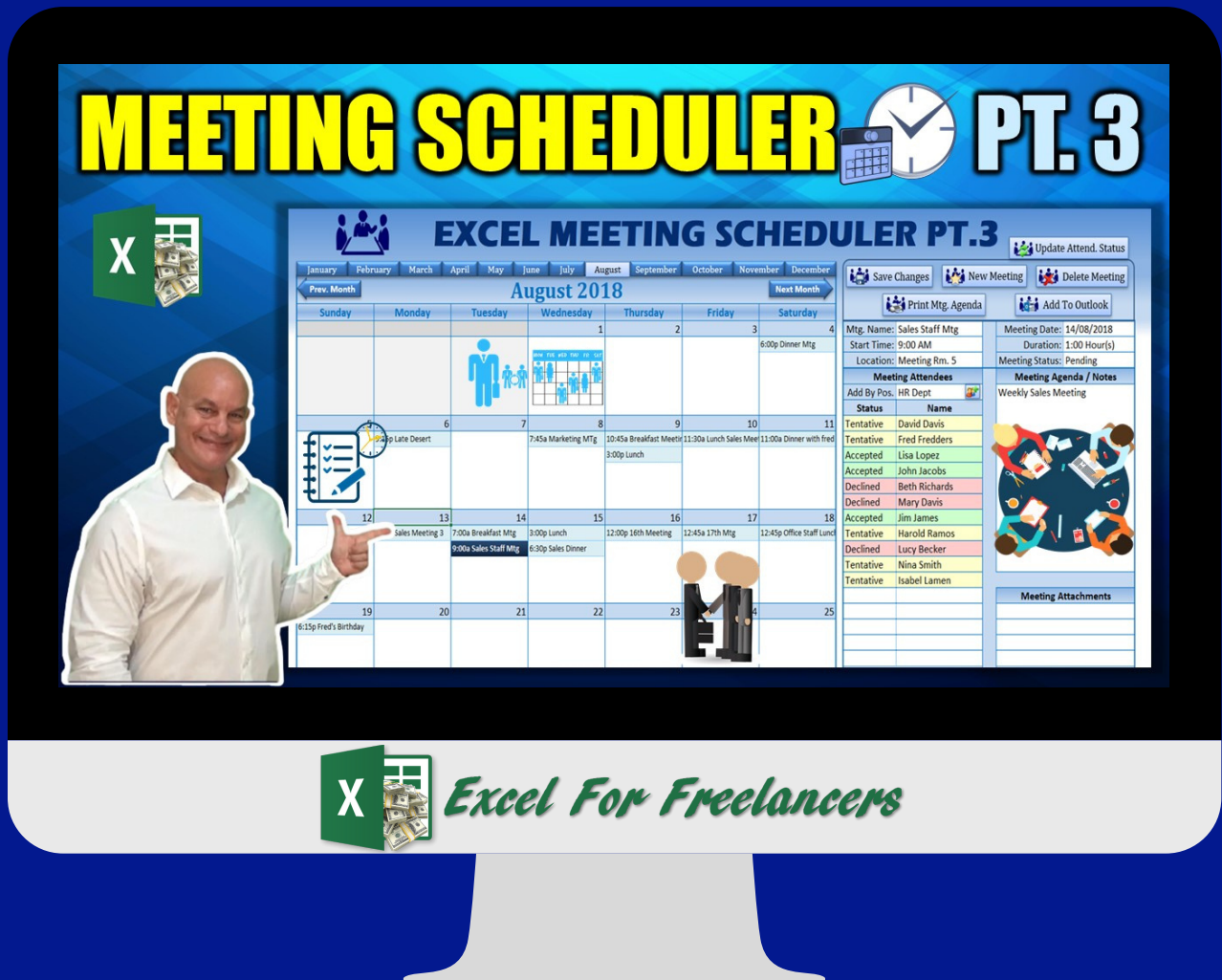


VBA SOURCE CODE BOOK



**Automate Attendee Responses &
Sync With Outlook with this
Excel Meeting Scheduler [Part 3]**



**DOWNLOAD
APPLICATION**



**WATCH ENTIRE
SERIES**



**VIEW
TRAINING**

by: Randy Austin

ABOUT THE AUTHOR

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

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

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



[WEBSITE](#)



[YOUTUBE](#)



[FACEBOOK](#)



[TWITTER](#)



[INSTAGRAM](#)



[TELEGRAM](#)



[RUMBLE](#)



Microsoft®
Most Valuable
Professional



OUR COURSES & PRODUCTS



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

[Click here to learn more](#)



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

[Click here to learn more](#)



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

[Click here to learn more](#)



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

[Click here to learn more](#)

Projects.....	4
VBAProject	4
Documents	4
Sheet1.....	4
(Declarations).....	4
Worksheet_BeforeDoubleClick [Sub]	4
Worksheet_Change [Sub]	4
Worksheet_Deactivate [Sub]	4
Worksheet_SelectionChange [Sub]	4
Sheet2.....	7
(Declarations).....	7
Sheet3.....	9
(Declarations).....	9
Worksheet_Deactivate [Sub]	9
Worksheet_SelectionChange [Sub]	9
Sheet4.....	11
(Declarations).....	11
Sheet5.....	13
(Declarations).....	13
Worksheet_SelectionChange [Sub]	13
Sheet6.....	15
(Declarations).....	15
Sheet7.....	17
(Declarations).....	17
Sheet8.....	19
(Declarations).....	19
Sheet9.....	21
(Declarations).....	21
ThisWorkbook	23
(Declarations).....	23
Workbook_Open [Sub]	23
Modules.....	25
AddToOutlook_Mac.....	25
(Declarations).....	25
AddApptToOutlook [Sub]	25
AttendeesMacros	27
(Declarations).....	27
AttendeeStatus_Refresh [Sub]	27
UpdateAttendeeStatus [Sub]	27
CodeReset_Macs	30
(Declarations).....	30
ResetCalc [Sub]	30
StopCalc [Sub]	30
Email_LoadSave	32
(Declarations).....	32
Email_Load [Sub]	32
Email_Save [Sub]	32
Email_SchedSendMac	34
(Declarations).....	34
Email_ScheduleSend [Sub]	34
Email_SendSchedNow [Sub]	34
Email_Send.....	37
(Declarations).....	37
SendEmail [Sub]	37
Email_SetTypes	40
(Declarations).....	40
Email_MeetingChange [Sub]	40
Email_MtgStatusChange [Sub]	40
Email_SelectedMeeting [Sub]	40
Email_SendAfterMtg [Sub]	41
Email_SendBeforeMtg [Sub]	42
FadeOutMessage.....	44
(Declarations).....	44
EmailSaved_Mesg [Sub]	44
MeetingSaved_Mesg [Sub]	44
MonthSel_Mac	46

Table of Contents

(Declarations).....	46
MonthSel1 [Sub]	46
MonthSel10 [Sub]	46
MonthSel11 [Sub]	46
MonthSel12 [Sub]	46
MonthSel2 [Sub]	46
MonthSel3 [Sub]	46
MonthSel4 [Sub]	46
MonthSel5 [Sub]	46
MonthSel6 [Sub]	46
MonthSel7 [Sub]	47
MonthSel8 [Sub]	47
MonthSel9 [Sub]	47
Mtg_Load	49
(Declarations).....	49
Meeting_Load [Sub]	49
Mtg_Macros	52
(Declarations).....	52
Mtg_AddAllByPosition [Sub]	52
Mtg_DeleteAttachment [Sub]	52
Mtg_DeleteAttendee [Sub]	52
Mtg_New_Delete	54
(Declarations).....	54
Meeting_Delete [Sub]	54
Meeting_New [Sub]	54
OutlookTest [Sub]	54
Mtg_SaveUpdate.....	56
(Declarations).....	56
SaveUpdate_Meeting [Sub]	56
PopUp_Calendar	59
(Declarations).....	59
CalCol1 [Sub]	59
CalCol2 [Sub]	59
CalCol3 [Sub]	59
CalCol4 [Sub]	59
CalCol5 [Sub]	59
CalCol6 [Sub]	59
CalCol7 [Sub]	59
CalCol8 [Sub]	59
CalCol9 [Sub]	60
CalendarHide [Sub]	60
CalendarShow [Sub]	60
CalFormulaReplacement [Sub]	61
CheckForSheet [Sub]	61
CreateCalSht [Sub]	61
DayBtn1 [Sub]	63
DayBtn10 [Sub]	63
DayBtn11 [Sub]	64
DayBtn12 [Sub]	64
DayBtn13 [Sub]	64
DayBtn14 [Sub]	64
DayBtn15 [Sub]	64
DayBtn16 [Sub]	64
DayBtn17 [Sub]	64
DayBtn18 [Sub]	65
DayBtn19 [Sub]	65
DayBtn2 [Sub]	65
DayBtn20 [Sub]	65
DayBtn21 [Sub]	65
DayBtn22 [Sub]	65
DayBtn23 [Sub]	65
DayBtn24 [Sub]	66
DayBtn25 [Sub]	66
DayBtn26 [Sub]	66
DayBtn27 [Sub]	66
DayBtn28 [Sub]	66

DayBtn29 [Sub]	66
DayBtn3 [Sub]	66
DayBtn30 [Sub]	67
DayBtn31 [Sub]	67
DayBtn32 [Sub]	67
DayBtn33 [Sub]	67
DayBtn34 [Sub]	67
DayBtn35 [Sub]	67
DayBtn36 [Sub]	67
DayBtn37 [Sub]	68
DayBtn38 [Sub]	68
DayBtn39 [Sub]	68
DayBtn4 [Sub]	68
DayBtn40 [Sub]	68
DayBtn41 [Sub]	68
DayBtn42 [Sub]	69
DayBtn5 [Sub]	69
DayBtn6 [Sub]	69
DayBtn7 [Sub]	69
DayBtn8 [Sub]	69
DayBtn9 [Sub]	69
GroupCal [Sub]	69
MacroLinkRemover [Sub]	70
NextMonth [Sub]	71
PrevMonth [Sub]	71
ShowSettings [Sub]	71
UnGroupCal [Sub]	72
SchedMacs	74
(Declarations)	74
AddAttachment [Sub]	74
LoadMonth [Sub]	74
NextMonth_Sched [Sub]	75
PrevMonth_Sched [Sub]	75
PrintMeeting [Sub]	75
StaffMacros	77
(Declarations)	77
ClearAdvancedFilter [Sub]	77
Staff_GetUniquePositions [Sub]	77

```

1 Option Explicit
2
3 Private Sub Worksheet_BeforeDoubleClick(ByVal Target As Range, Cancel As Boolean)
4     If Not Intersect(Target, Range("L11:L40" )) Is Nothing Then
5         If Target.Value = "Tentative" Then Target.Value = "Accepted"
6         Cancel = True
7     End If
8 End Sub
9
10 Private Sub Worksheet_Change(ByVal Target As Range)
11     If Target.Count > 2 Then Exit Sub
12     If Not Intersect(Target, Range("M11:M40" )) Is Nothing And Range("B2" ).Value = False Then
13         'On Change of Attendee But not on Meeting Load
14         If Target.Value <> Empty And Range("L" & Target.Row).Value = Empty Then Range("L" &
15             Target.Row).Value = "Tentative"
16     End If
17     If Not Intersect(Target, Range("M5:M7,P5:P6" )) Is Nothing And Range("B2" ).Value = False
18     Then 'Mark Meeting Fields as Changed
19         If Target.Column = 13 Then Range("B" & Target.Row + 6).Value = True
20         If Target.Column = 16 Then Range("B" & Target.Row + 9).Value = True
21     End If
22     If Not Intersect(Target, Range("P7" )) Is Nothing And Range("B2" ).Value = False Then 'On
23     Status Change
24     Range("B" & Range("B21" ).Value).Value = True 'Set Meeting Status to true
25 End If
26 If Not Intersect(Target, Range("S4,S5" )) Is Nothing Then LoadMonth
27 End Sub
28
29 Private Sub Worksheet_Deactivate()
30     Dim Cal As Shape
31     On Error Resume Next
32     Set Cal = Shapes("Calendar" )
33     On Error GoTo 0
34     If Cal Is Nothing Then 'Shape Deleted
35         Application.EnableEvents = False
36         On Error Resume Next
37         Application.Undo
38         On Error GoTo 0
39         Application.EnableEvents = True
40     End If
41 End Sub
42
43 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
44     If Target.Count > 2 Then Exit Sub
45     Dim Cal As Shape
46     On Error Resume Next
47     Set Cal = Shapes("Calendar" )
48     On Error GoTo 0
49     If Cal Is Nothing Then 'Shape Deleted
50         Application.EnableEvents = False
51         On Error Resume Next
52         Application.Undo
53         On Error GoTo 0
54         Application.EnableEvents = True
55     End If
56     If Not Intersect(Target, Range("P5" )) Is Nothing Then 'Change this to any cell (or
57     cells) you would like to have the Pop-Up Calendar Appear
58     CheckForSheet
59     CalendarShow
60 Else:
61     CheckForSheet
62     CalendarHide

```



```
57     1 2 End If
58     If Not Intersect(Target, Range("D6:J10,D12:J16,D18:J22,D24:J28,D30:J34,D36:J40" )) Is
Nothing Then
59         If Target.Value <> Empty Then
60             Range("B3" ).Value = Cells(Target.Row, Target.Column + 50).Value
61             If Range("B3" ).Value <> Empty Then Meeting_Load
62         End If
63     End If
64
65     If Not Intersect(Target, Range("O23:P40" )) Is Nothing Then
66         With Shapes("AddFileBtn" )
67             .Visible = msoCTrue
68             .Left = Range("Q" & Target.Row).Left
69             .Top = Range("Q" & Target.Row).Top
70         End With
71         If Range("O" & Target.Row).Value <> Empty Then
72             With Shapes("DeleteAttachBtn" )
73                 .Visible = msoCTrue
74                 .Left = Range("Q" & Target.Row).Left
75                 .Top = Range("Q" & Target.Row).Top
76                 .IncrementLeft -16
77             End With
78         Else:
79             Shapes("DeleteAttachBtn" ).Visible = msoFalse
80         End If
81     Else:
82         Shapes("AddFileBtn" ).Visible = msoFalse
83         Shapes("DeleteAttachBtn" ).Visible = msoFalse
84     End If
85     If Not Intersect(Target, Range("M11:M40" )) Is Nothing And Range("M" & Target.Row).Value
<> Empty Then
86         With Shapes("DeleteAttendBtn" )
87             .Visible = msoCTrue
88             .Left = Range("N" & Target.Row).Left
89             .Top = Range("N" & Target.Row).Top
90             .IncrementLeft -16
91         End With
92     Else:
93         Shapes("DeleteAttendBtn" ).Visible = msoFalse
94     End If
95
96 End Sub
97
98
99
```


A

Application, 4

C

Cal, 4

CalendarHide, 4

CalendarShow, 4

Cancel, 4

Cells, 5

CheckForSheet, 4

Column, 4, 5

Count, 4

E

Empty, 4, 5

EnableEvents, 4

Explicit, 4

I

IncrementLeft, 5

Intersect, 4, 5

L

Left, 5

LoadMonth, 4

M

Meeting_Load, 5

msoCTrue, 5

msoFalse, 5

R

Range, 4, 5

Row, 4, 5

S

Shape, 4

Shapes, 4, 5

T

Target, 4, 5

Top, 5

U

Undo, 4

V

Value, 4, 5

Visible, 5

W

Worksheet_BeforeDoubleClick, 4

Worksheet_Change, 4

Worksheet_Deactivate, 4

Worksheet_SelectionChange, 4

1 Option Explicit

2

E

Explicit, 7

```
1 Option Explicit
2
3 Private Sub Worksheet_Deactivate()
4     Staff_GetUniquePositions
5 End Sub
6
7 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
8     If Target.Count > 1 Then Exit Sub
9     If Not Intersect(Target, Range("C3:F53" )) Is Nothing Then
10        Range("B2" ).Value = Target.Row
11    End If
12 End Sub
```

C

Count, [9](#)

E

Explicit, [9](#)

I

Intersect, [9](#)

R

Range, [9](#)

Row, [9](#)

S

Staff_GetUniquePositions, [9](#)

T

Target, [9](#)

V

Value, [9](#)

W

Worksheet_Deactivate, [9](#)

Worksheet_SelectionChange, [9](#)

1 Option Explicit

2

E

Explicit, [11](#)


```
1 Option Explicit
2
3 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
4     If Target.Count > 1 Then Exit Sub
5     If Not Intersect(Target, Range("D21:V27" )) Is Nothing Then
6         Range("B4" ).Value = Target.Row
7         Email_Load
8     End If
9 End Sub
```

C

Count, [13](#)

E

Email_Load, [13](#)

Explicit, [13](#)

I

Intersect, [13](#)

R

Range, [13](#)

Row, [13](#)

T

Target, [13](#)

V

Value, [13](#)

W

Worksheet_SelectionChange, [13](#)

1 Option Explicit

2

E

Explicit, [15](#)

1 Option Explicit

2

E

Explicit, [17](#)

1 Option Explicit

2

E

Explicit, [19](#)

1 Option Explicit

2

E

Explicit, [21](#)

```
1 Option Explicit
2
3 Private Sub Workbook_Open()
4     'Email_ScheduleSend ' Un-comment this macro to check for scheduled emails on workbook
5     open
6 End Sub
```

E

Explicit, [23](#)

W

Workbook_Open, [23](#)

```
1 Option Explicit
2
3 Sub AddApptToOutlook()
4     Dim OutApp, OutItem, OutAttendee, OutCont As Object
5     Set OutApp = CreateObject("Outlook.Application" )
6     Set OutItem = OutApp.CreateItem(olAppointmentItem)
7     Dim MtgDate As Date
8     With Sheet1
9         MtgDate = .Range("P5" ).Value 'Meeting Date
10
11         With OutItem
12             .MeetingStatus = olMeeting
13             .Subject = Sheet1.Range("M5" ).Value 'Meeting Name
14             .Body = Sheet1.Range("O9" ).Value 'Meeting Agenda/Notes
15             .Location = Sheet1.Range("M7" ).Value
16             .Start = MtgDate + Sheet1.Range("M6" ).Value
17             .End = MtgDate + Sheet1.Range("M6" ).Value + Sheet1.Range("P6" ).Value
18         End With
19
20         OutItem.Display
21     End With
22 End Sub
```

A

AddApptToOutlook, [25](#)

B

Body, [25](#)

C

CreateItem, [25](#)

CreateObject, [25](#)

D

Display, [25](#)

E

Explicit, [25](#)

L

Location, [25](#)

M

MeetingStatus, [25](#)

MtgDate, [25](#)

O

olAppointmentItem, [25](#)

olMeeting, [25](#)

OutApp, [25](#)

OutAttendee, [25](#)

OutCont, [25](#)

OutItem, [25](#)

R

Range, [25](#)

S

Sheet1, [25](#)

Start, [25](#)

Subject, [25](#)

V

Value, [25](#)

Option Explicit

```

1  Option Explicit
2
3  Sub AttendeeStatus_Refresh()
4      Dim AttStatusLink As String
5      If Sheet1.Range("S7" ).Value = Empty Then
6          MsgBox "Please enter a Google Sheets link"
7          Exit Sub
8      End If
9      AttStatusLink = Sheet1.Range("S7" ).Value
10     Sheet9.Range("B4:D9999" ).ClearContents
11     'Refresh Attendee Status
12     On Error Resume Next
13     ActiveWorkbook.Connections("Status_Connection" ).Delete
14     On Error GoTo 0
15     On Error Resume Next
16     Sheet9.Names("Status_Connection" ).Delete
17     On Error GoTo 0
18     With Sheet9.QueryTables.Add(Connection:="URL;" & AttStatusLink, Destination:=Sheet9.Range
19         ("A$1" ))
20         .Name = "Status_Connection"
21         .FieldNames = True
22         .RowNumbers = False
23         .FillAdjacentFormulas = False
24         .PreserveFormatting = False
25         .RefreshOnFileOpen = False
26         .BackgroundQuery = True
27         .RefreshStyle = xlOverwriteCells
28         .SavePassword = False
29         .SaveData = True
30         .AdjustColumnWidth = True
31         .RefreshPeriod = 0
32         .WebSelectionType = xlEntirePage
33         .WebFormatting = xlWebFormattingAll
34         .WebPreFormattedTextToColumns = True
35         .WebConsecutiveDelimitersAsOne = True
36         .WebSingleBlockTextImport = False
37         .WebDisableDateRecognition = False
38         .WebDisableRedirections = False
39         .Refresh BackgroundQuery:=False
40     End With
41     UpdateAttendeeStatus
42     Meeting_Load
43 End Sub
44
45 Sub UpdateAttendeeStatus()
46     Dim LastUpdateRow, LastStatusRow, StatusDbRow, AttendDbRow As Long
47     LastUpdateRow = Sheet1.Range("B1" ).Value
48     LastStatusRow = Sheet9.Range("B99999" ).End(xlUp).Row
49     If LastStatusRow = LastUpdateRow Then Exit Sub 'No New Rows
50     If LastStatusRow <= LastUpdateRow Then LastStatusRow = LastUpdateRow + 1
51     Sheet9.Range("F" & LastUpdateRow + 1 & ":F" & LastStatusRow).Formula =
52         "=IFERROR(MATCH(C" & LastUpdateRow + 1 & ",AttendMtgIDName,0)+3,)"
53     Sheet9.Calculate
54     For StatusDbRow = LastUpdateRow + 1 To LastStatusRow
55         If Sheet9.Range("F" & StatusDbRow).Value <> "" Then
56             AttendDbRow = Sheet9.Range("F" & StatusDbRow).Value
57             'On Error Resume Next
58             Sheet4.Range("E" & AttendDbRow).Value = Sheet9.Range("E" & StatusDbRow).Value
59             'Update Status
60         End If
61     Next StatusDbRow

```

1

```
59 | 1 Sheet1.Range("B1" ).Value = LastStatusRow  
60 | End Sub
```

A

ActiveWorkbook, [27](#)
Add, [27](#)
AdjustColumnWidth, [27](#)
AttendDbRow, [27](#)
AttendeeStatus_Refresh, [27](#)
AttStatusLink, [27](#)

B

BackgroundQuery, [27](#)

C

Calculate, [27](#)
ClearContents, [27](#)
Connection, [27](#)
Connections, [27](#)

D

Delete, [27](#)
Destination, [27](#)

E

Empty, [27](#)
Explicit, [27](#)

F

FieldNames, [27](#)
FillAdjacentFormulas, [27](#)
Formula, [27](#)

L

LastStatusRow, [27](#), [28](#)
LastUpdateRow, [27](#)

M

Meeting_Load, [27](#)
MsgBox, [27](#)

N

Name, [27](#)
Names, [27](#)

P

PreserveFormatting, [27](#)

Q

QueryTables, [27](#)

R

Range, [27](#), [28](#)
Refresh, [27](#)
RefreshOnFileOpen, [27](#)
RefreshPeriod, [27](#)
RefreshStyle, [27](#)
Row, [27](#)
RowNumbers, [27](#)

S

SaveData, [27](#)
SavePassword, [27](#)
Sheet1, [27](#), [28](#)
Sheet4, [27](#)
Sheet9, [27](#)
StatusDbRow, [27](#)

U

UpdateAttendeeStatus, [27](#)

V

Value, [27](#), [28](#)

W

WebConsecutiveDelimitersAsOne, [27](#)
WebDisableDateRecognition, [27](#)
WebDisableRedirections, [27](#)
WebFormatting, [27](#)
WebPreFormattedTextToColumns, [27](#)
WebSelectionType, [27](#)
WebSingleBlockTextImport, [27](#)

X

xlEntirePage, [27](#)
xlOverwriteCells, [27](#)
xlUp, [27](#)
xlWebFormattingAll, [27](#)

```
1 Option Explicit
2
3 Sub ResetCalc()
4     With Application
5         .EnableEvents = True
6         .Calculation = xlCalculationAutomatic
7         .ScreenUpdating = True
8     End With
9 End Sub
10
11 Sub StopCalc()
12     With Application
13         .EnableEvents = False
14         .Calculation = xlCalculationManual
15         .ScreenUpdating = False
16     End With
17 End Sub
18
```

A

Application, [30](#)

C

Calculation, [30](#)

E

EnableEvents, [30](#)

Explicit, [30](#)

R

ResetCalc, [30](#)

S

ScreenUpdating, [30](#)

StopCalc, [30](#)

X

xlCalculationAutomatic, [30](#)

xlCalculationManual, [30](#)

```
1 Option Explicit
2
3 Sub Email_Load()
4     Dim EmailCol, EmailRow As Long
5     Dim CellRng, Trigger As String
6     StopCalc
7     With Sheet5
8         EmailRow = .Range("B4").Value
9         For EmailCol = 4 To 24
10            CellRng = .Cells(18, EmailCol).Value
11            .Range(CellRng).Value = .Cells(EmailRow, EmailCol).Value
12        Next EmailCol
13
14        'Show/Hide Options
15        Trigger = .Range("E4").Value 'Trigger Type
16        .Shapes("MtgChangeGrp").Visible = msoFalse
17        .Shapes("MtgStatusGrp").Visible = msoFalse
18        .Shapes("AttendStatusGrp").Visible = msoFalse
19
20        If Trigger = "Meeting Changed" Then .Shapes("MtgChangeGrp").Visible = msoCTrue
21        If Trigger = "Attendee Status Change" Then .Shapes("AttendStatusGrp").Visible =
22            msoCTrue
23        If Trigger = "Meeting Status Change" Then .Shapes("MtgStatusGrp").Visible = msoCTrue
24    End With
25    ResetCalc
26 End Sub
27
28 Sub Email_Save()
29     Dim EmailCol, EmailRow, SheetRow As Long
30     StopCalc
31     With Sheet5
32         EmailRow = .Range("B4").Value
33         'Add in Options
34         For SheetRow = 8 To 27
35             EmailCol = .Range("C" & SheetRow)
36             If .Range("C" & SheetRow).Value <> Empty Then .Cells(EmailRow, EmailCol).Value = .
37                 Range("B" & SheetRow).Value
38         Next SheetRow
39         .Range("F" & EmailRow).Value = .Range("F4").Value 'Qty
40         .Range("G" & EmailRow).Value = .Range("G4").Value 'Days/Hours
41         .Range("H" & EmailRow).Value = .Range("E7").Value 'Subject
42         .Range("I" & EmailRow).Value = .Range("E8").Value 'Message
43         .Range("L" & EmailRow).Value = .Range("E6").Value 'CC
44
45         .Range("H20:L27").WrapText = False
46     End With
47     ResetCalc
48     EmailSaved_Mesg
49 End Sub
```

C

CellRng, [32](#)
Cells, [32](#)

E

Email_Load, [32](#)
Email_Save, [32](#)
EmailCol, [32](#)
EmailRow, [32](#)
EmailSaved_Mesg, [32](#)
Empty, [32](#)
Explicit, [32](#)

M

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

R

Range, [32](#)
ResetCalc, [32](#)

S

Shapes, [32](#)
Sheet5, [32](#)
SheetRow, [32](#)
StopCalc, [32](#)

T

Trigger, [32](#)

V

Value, [32](#)
Visible, [32](#)

W

WrapText, [32](#)

Option Explicit

Sub Email_ScheduleSend()

```
Dim LastLogRow, EmailLogRow, LastLogFiltRow, LogFiltRow As Long
```

```
With Sheet8
```

```
.Range("AA2:AH2,AA4:AH9999").ClearContents 'Clear Prev Advanced Filters & Results
```

```
.Range("AG2").Value = "=" 'Set Criteria for all blank Email Sent
```

```
LastLogRow = .Range("D9999").End(xlUp).Row 'Last Log Row
```

```
.Range("D5:K" & LastLogRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range("AA1:AH2"), CopyToRange:=.Range("AA3:AH3"), unique:=True
```

```
LastLogFiltRow = .Range("AA999").End(xlUp).Row
```

```
If LastLogFiltRow < 4 Then GoTo NoEmails
```

```
'Sort Send On Ascending
```

```
.Sort.SortFields.Clear 'Clear any existing sorts
```

```
.Sort.SortFields.Add Key:=Sheet6.Range("AF3"), SortOn:=xlSortOnValues, Order:=xlAscending, DataOption:=xlSortNormal 'Sort By Time
```

```
With Sheet8.Sort
```

```
.SetRange Range("AA4:AH" & LastLogFiltRow)
```

```
.Apply
```

```
End With
```

```
LogFiltRow = 4 'Earliest Scheduled Send Email
```

```
.Range("B7").Value = .Range("AA" & LogFiltRow).Value 'Set Meeting ID
```

```
.Range("B8").Value = .Range("AB" & LogFiltRow).Value 'Set Trigger
```

```
.Range("B9").Value = .Range("AH" & LogFiltRow).Value 'Set Log Row
```

```
If .Range("AF" & LogFiltRow).Value < Now Then 'Check if in the past or not  
Email_SendSchedNow
```

```
Else: 'if not, schedule to run in the future
```

```
Application.OnTime .Range("AF" & LogFiltRow).Value, "Email_SendSchedNow"
```

```
End If
```

```
End With
```

```
NoEmails:
```

```
End Sub
```

Sub Email_SendSchedNow()

```
Dim MtgID, Trigger As String
```

```
Dim MtgRow, LogRow As Long
```

```
Dim LastAttendRow, LastAttendFiltRow As Long
```

```
Dim LastAttachRow, LastAttachFiltRow As Long
```

```
Dim MtgIDRng, FoundMtgIDRng As Range
```

```
Set MtgIDRng = Sheet6.Range("MtgID")
```

```
MtgID = Sheet8.Range("B7").Value 'Meeting ID
```

```
Trigger = Sheet8.Range("B8").Value 'Trigger
```

```
LogRow = Sheet8.Range("B9").Value 'Log Row
```

```
Set FoundMtgIDRng = MtgIDRng.Find(MtgID, , xlValues, xlWhole)
```

```
If Not FoundMtgIDRng Is Nothing Then 'Found Meeting ID
```

```
MtgRow = FoundMtgIDRng.Row
```

```
Sheet5.Range("B30:B40,A44:A73,A76:A93").ClearContents 'Clear Prev Email Fields
```

```
With Sheet5
```

```
If Trigger = "Send After Meeting" Then .Range("B30").Value = 26 Else: .Range("B30").Value = 27 'Set Email Trigger Row
```

```
.Range("B31").Value = MtgID 'Meeting ID
```

```
.Range("B32").Value = Sheet6.Range("D" & MtgRow).Value 'Meeting Name
```

```
.Range("B33").Value = Sheet6.Range("E" & MtgRow).Value 'Date
```

```
.Range("B34").Value = Sheet6.Range("F" & MtgRow).Value 'Start Time
```

```
.Range("B35").Value = Sheet6.Range("H" & MtgRow).Value 'Location
```

```
.Range("B36").Value = Sheet6.Range("G" & MtgRow).Value 'Duration
```

```
.Range("B38").Value = Sheet6.Range("I" & MtgRow).Value 'Mtg Status
```

```
.Range("B39").Value = Sheet6.Range("J" & MtgRow).Value 'Agenda
```



```

59      1  2  3
60      'Get Attendees
61      LastAttendRow = Sheet4.Range("C99999" ).End(xlUp).Row
62      If LastAttendRow < 4 Then GoTo NoAttend
63      Sheet4.Range("AA2:AE2,AA4:AE99999" ).ClearContents 'Clear Remaing Adv. Filter
        Criteria & Results
64      Sheet4.Range("AA2" ).Value = MtgID
65      Sheet4.Range("C3:G" & LastAttendRow).AdvancedFilter xlFilterCopy, CriteriaRange:=
        Sheet4.Range("AA1:AE2" ), CopyToRange:=Sheet4.Range("AB3:AE3" ), unique:=True
66      LastAttendFiltRow = Sheet4.Range("AC99" ).End(xlUp).Row
67      If LastAttendFiltRow < 3 Then GoTo NoAttend
68      .Range("A44:A" & LastAttendFiltRow + 40).Value = Sheet4.Range("AC4:AC" &
        LastAttendFiltRow).Value 'Copy Over Attendees
69      NoAttend:
70
71
72      'Get Attachments
73      LastAttachRow = Sheet2.Range("C99999" ).End(xlUp).Row
74      If LastAttachRow < 4 Then GoTo NoAttach
75      Sheet2.Range("AA2:AE2,AA4:AE99999" ).ClearContents 'Clear Remaing Adv. Filter
        Criteria & Results
76      Sheet2.Range("AA2" ).Value = MtgID
77      Sheet2.Range("C3:G" & LastAttachRow).AdvancedFilter xlFilterCopy, CriteriaRange:=
        Sheet2.Range("AA1:AE2" ), CopyToRange:=Sheet2.Range("AB3:AE3" ), unique:=True
78      LastAttachFiltRow = Sheet2.Range("AB99" ).End(xlUp).Row
79      If LastAttachFiltRow < 4 Then GoTo NoAttach
80      .Range("A76:A" & LastAttachFiltRow + 72).Value = Sheet2.Range("AC4:AC" &
        LastAttachFiltRow).Value 'File Path
81      NoAttach:
82
83      SendEmail 'Run Send Email Macro
84      Sheet8.Range("J" & LogRow).Value = Now 'Set Email Sent
85
86      End With
87      } Else: 'Meeting not Found (perhaps deleted), add N/A
88      Sheet8.Range("J" & Sheet8.Range("AH4" ).Value).Value = "N/A"
89      End If 'Found Mtg Row
90      Email_ScheduleSend 'Rerun & Schedule Next Email
91      End Sub

```

A

Add, 34
AdvancedFilter, 34, 35
Application, 34
Apply, 34

C

Clear, 34
ClearContents, 34, 35
CopyToRange, 34, 35
CriteriaRange, 34, 35

D

DataOption, 34

E

Email_ScheduleSend, 34, 35
Email_SendSchedNow, 34
EmailLogRow, 34
Explicit, 34

F

Find, 34
FoundMtgIDRng, 34

K

Key, 34

L

LastAttachFiltRow, 34, 35
LastAttachRow, 34, 35
LastAttendFiltRow, 34, 35
LastAttendRow, 34, 35
LastLogFiltRow, 34
LastLogRow, 34
LogFiltRow, 34
LogRow, 34, 35

M

MtgID, 34, 35
MtgIDRng, 34
MtgRow, 34

N

NoAttach, 35
NoAttend, 35
NoEmails, 34
Now, 34, 35

O

OnTime, 34
Order, 34

R

Range, 34, 35
Row, 34, 35

S

SendEmail, 35
SetRange, 34
Sheet2, 35
Sheet4, 35
Sheet5, 34
Sheet6, 34
Sheet8, 34, 35
Sort, 34
SortFields, 34
SortOn, 34

T

Trigger, 34

U

unique, 34, 35

V

Value, 34, 35

X

xlAscending, 34
xlFilterCopy, 34, 35
xlSortNormal, 34
xlSortOnValues, 34
xlUp, 34, 35
xlValues, 34
xlWhole, 34

```

1 Option Explicit
2
3 Sub SendEmail()
4 Dim MtgEmailRow, MtgName, Location, Duration, StartTime, AttendName, MtgStatus, Agenda,
AllAttendees As String
5 Dim CCEmail, EmailAdd, Subj, Mesg, AcceptLink, MtgID, DeclineLink As String
6 Dim LastEmailAddRow, EmailAddRow, LastAttachRow, AttachRow As Long
7 Dim SchedDate As Date
8 Dim OutApp, OutMail As Object
9 Set OutApp = CreateObject("Outlook.Application" )
10 With Sheet5
11 .Calculate
12 MtgID = .Range("B31" ).Value 'Meeting ID
13 MtgEmailRow = .Range("B30" ).Value 'Meeting Trigger Row
14 MtgName = .Range("B32" ).Value 'Meeting Name
15 SchedDate = .Range("B33" ).Value 'Schedule Date
16 StartTime = Application.WorksheetFunction.Text(.Range("B34" ).Value, "h:mm AM/PM" )
'Start Time
17 Location = .Range("B35" ).Value 'Location
18 MtgStatus = .Range("B38" ).Value 'Meeting Status
19 Agenda = .Range("B39" ).Value 'Agenda
20 AllAttendees = .Range("B40" ).Value 'All Attendees List
21 CCEmail = .Range("L" & MtgEmailRow).Value 'Email Subject
22
23
24 If .Range("B36" ).Value < 0.0416666 Then 'Set Duration to Hours or Minutes
25 Duration = Application.WorksheetFunction.Text(.Range("B36" ).Value, "h:mm " & Chr(
34) & "minutes" & Chr(34)) 'Duration
26 Else:
27 Duration = Application.WorksheetFunction.Text(.Range("B36" ).Value, "h:mm " & Chr(
34) & "hour(s)" & Chr(34)) 'Duration
28 End If
29
30 LastEmailAddRow = .Range("A74" ).End(xlUp).Row 'Last Email Address Row
31 If LastEmailAddRow < 44 Then GoTo NoEmail 'Check to ensure emails
32 For EmailAddRow = 44 To LastEmailAddRow 'Run Loop Through Emails
33
34 'Add in Subject & Message Variables
35 Subj = .Range("H" & MtgEmailRow).Value 'Subject
36 Mesg = .Range("I" & MtgEmailRow).Value 'Email Message
37 AttendName = .Range("A" & EmailAddRow).Value 'Attend Name
38 EmailAdd = .Range("B" & EmailAddRow).Value 'Attend Email
39 AcceptLink = "<a href=" & Chr(34) & Sheet1.Range("S6" ).Value & "?MtgID=" & MtgID
& "&" & "ID_Name=" & MtgID & "_" & AttendName & "&" & "Name=" & AttendName &
&" & "Status=Accepted" & Chr(34) & ">ACCEPT</a>"
40 DeclineLink = "<a href=" & Chr(34) & Sheet1.Range("S6" ).Value & "?MtgID=" & MtgID
& "&" & "ID_Name=" & MtgID & "_" & AttendName & "&" & "Name=" & AttendName &
&" & "Status=Declined" & Chr(34) & ">DECLINE</a>"
41 Subj = Replace(Replace(Replace(Replace(Replace(Replace(Replace(Replace(Replace(
Replace(Replace(Replace(Subj, _
42 "#MtgName#" , MtgName), "#SchedDate#" , SchedDate), "#StartTime#" ,
StartTime), _
43 "#Location#" , Location), "#Duration#" , Duration), "#MtgStatus#" ,
MtgStatus), _
44 "#Agenda#" , Agenda), "#AllAttendees#" , AllAttendees), "#AttendName#" ,
AttendName), _
45 "#AcceptLink#" , AcceptLink), "#DeclineLink#" , DeclineLink), Chr(10),
"<br>" )
46 Mesg = Replace(Replace(Replace(Replace(Replace(Replace(Replace(Replace(Replace(
Replace(Replace(Replace(Mesg, _
47 "#MtgName#" , MtgName), "#SchedDate#" , SchedDate), "#StartTime#" ,

```

1 2 3

```

1 2 3
48     StartTime), _
      "#Location#" , Location), "#Duration#" , Duration), "#MtgStatus#" ,
49     MtgStatus), _
      "#Agenda#" , Agenda), "#AllAttendees#" , AllAttendees), "#AttendName#" ,
50     AttendName), _
      "#AcceptLink#" , AcceptLink), "#DeclineLink#" , DeclineLink), Chr(10),
      "<br>" )
51
52     Set OutMail = OutApp.CreateItem(0)
53     With OutMail
54         .To = EmailAdd
55         .CC = CCEmail
56         .Subject = Subj
57         .HTMLBody = Mesg
58         'Check for Add Attach Option AND existing Attachments
59         If Sheet5.Range("K" & MtgEmailRow).Value = True And Sheet5.Range("A76" ).Value <
            > Empty Then
60             LastAttachRow = Sheet5.Range("A93" ).End(xlUp).Row 'Set Last Attachment Row
61             For AttachRow = 76 To LastAttachRow
62                 If Sheet5.Range("A" & AttachRow).Value <> Empty Then
63                     .Attachments.Add Sheet5.Range("A" & AttachRow).Value
64                 End If
65             Next AttachRow
66         End If
67         If Sheet5.Range("J" & MtgEmailRow).Value = True Then .Display Else: .Send
            'Dispaly or Send Based On Option
68     End With
69
70     Next EmailAddRow
71     NoEmail:
72     End With
73 End Sub

```

—, 37, 38

A

AcceptLink, 37, 38
Add, 38
Agenda, 37, 38
AllAttendees, 37, 38
Application, 37
Attachments, 38
AttachRow, 37, 38
AttendName, 37, 38

C

Calculate, 37
CC, 38
CCEmail, 37, 38
Chr, 37, 38
CreateItem, 38
CreateObject, 37

D

DeclineLink, 37, 38
Display, 38
Duration, 37, 38

E

EmailAdd, 37, 38
EmailAddRow, 37, 38
Empty, 38
Explicit, 37

H

HTMLBody, 38

L

LastAttachRow, 37, 38
LastEmailAddRow, 37
Location, 37, 38

M

Mesg, 37, 38
MtgEmailRow, 37, 38
MtgID, 37
MtgName, 37
MtgStatus, 37, 38

N

NoEmail, 37, 38

O

OutApp, 37, 38
OutMail, 37, 38

R

Range, 37, 38
Replace, 37
Row, 37, 38

S

SchedDate, 37
Send, 38
SendEmail, 37
Sheet1, 37
Sheet5, 37, 38
StartTime, 37, 38
Subj, 37, 38
Subject, 38

T

Text, 37

V

Value, 37, 38

W

WorksheetFunction, 37

X

xLUp, 37, 38

```

1  Option Explicit
2
3  Sub Email_MeetingChange()
4      If Sheet5.Range("B23").Value = True And Sheet1.Range("B11").Value = True Then GoTo
5          SendEmail 'Meeting Name Change
6      If Sheet5.Range("B24").Value = True And Sheet1.Range("B12").Value = True Then GoTo
7          SendEmail 'Meeting Start Time Change
8      If Sheet5.Range("B25").Value = True And Sheet1.Range("B13").Value = True Then GoTo
9          SendEmail 'Meeting Location Change
10     If Sheet5.Range("B26").Value = True And Sheet1.Range("B14").Value = True Then GoTo
11         SendEmail 'Meeting Date Change
12     If Sheet5.Range("B27").Value = True And Sheet1.Range("B15").Value = True Then GoTo
13         SendEmail ' Meeting Duration Change
14     Exit Sub 'Exit if none of the options above are true
15     SendEmail:
16     Sheet5.Range("B30").Value = 22 'Set Meeting Change Email Row
17     Email_SelectedMeeting
18 End Sub
19
20 Sub Email_MtgStatusChange()
21     If Sheet5.Range("B17").Value = True And Sheet1.Range("B17").Value = True Then GoTo
22         SendEmail 'Pending Status Change
23     If Sheet5.Range("B18").Value = True And Sheet1.Range("B18").Value = True Then GoTo
24         SendEmail 'Pending Status Change
25     If Sheet5.Range("B19").Value = True And Sheet1.Range("B19").Value = True Then GoTo
26         SendEmail 'Pending Status Change
27     If Sheet5.Range("B20").Value = True And Sheet1.Range("B20").Value = True Then GoTo
28         SendEmail 'Pending Status Change
29     Exit Sub 'Exit if none of the options above are true
30     SendEmail:
31     Sheet5.Range("B30").Value = 25 'Set Meeting Status Change Email Row
32     Email_SelectedMeeting
33 End Sub
34
35 Sub Email_SelectedMeeting()
36     'Email Settings set for New Meetings
37     Dim LastAttendRow, AttendRow, AttendEmailRow As Long
38     Dim LastAttachRow, AttachRow, AttachEmailRow As Long
39     LastAttendRow = Sheet1.Range("M40").End(xlUp).Row
40     LastAttachRow = Sheet1.Range("O40").End(xlUp).Row
41     AttendEmailRow = 44
42     AttachEmailRow = 76
43     With Sheet5
44         .Range("B32:B40,A45:A73,A76:A93").ClearContents 'Clear Prev. Email Values
45         If .Range("B30").Value <> 24 Then .Range("A44").ClearContents 'Clear Single Attenee
46             on Non Attendee Status Update
47         .Range("B31").Value = Sheet1.Range("B3").Value 'Meeting ID
48         .Range("B32").Value = Sheet1.Range("M5").Value 'Meeting Name
49         .Range("B33").Value = Sheet1.Range("P5").Value 'Date
50         .Range("B34").Value = Sheet1.Range("M6").Value 'Start Time
51         .Range("B35").Value = Sheet1.Range("M7").Value 'Location
52         .Range("B36").Value = Sheet1.Range("P6").Value 'Duration
53         .Range("B38").Value = Sheet1.Range("P7").Value 'Meeting Status
54         .Range("B39").Value = Sheet1.Range("O9").Value 'Agenda
55     End With
56     'Get All or Only New Attendees
57     If LastAttendRow < 11 Or .Range("B30").Value = 24 Then GoTo NoAttend 'If No Attendees
58         or on Attende Status Change, skip Attendees
59     For AttendRow = 11 To LastAttendRow
60         If Sheet1.Range("L" & AttendRow).Value <> Empty Then
61             If .Range("B30").Value <> 23 Then 'on Attendee added only add new Attendees,
62                 not all
63                 .Range("B40").Value = .Range("B40").Value & Sheet1.Range("M" & AttendRow).

```

1 2 3 4 5

```

1 2 3 4 5
50 Value & vbCrLf
   .Range("A" & AttendEmailRow).Value = Sheet1.Range("M" & AttendRow).Value
   'Add In Attendee Name
51 AttendEmailRow = AttendEmailRow + 1 'Increase Row By 1
52 } ElseIf Sheet1.Range("N" & AttendRow).Value = Empty Then
53   .Range("B40").Value = .Range("B40").Value & Sheet1.Range("M" & AttendRow).
   Value & vbCrLf
54   .Range("A" & AttendEmailRow).Value = Sheet1.Range("M" & AttendRow).Value
   'Add In Attendee Name
55   AttendEmailRow = AttendEmailRow + 1 'Increase Row By 1
56   End If
57 End If
58 Next AttendRow
59 NoAttend:
60
61 'Get All Attachments
62 If LastAttachRow < 23 Then GoTo NoAttach
63 For AttachRow = 23 To LastAttachRow
64   If Sheet1.Range("R" & AttachRow).Value <> Empty Then
65     .Range("A" & AttachEmailRow).Value = Sheet1.Range("R" & AttachRow).Value 'Add
     In Attachment File Path
66     AttachEmailRow = AttachEmailRow + 1 'Increase Row By 1
67   End If
68 Next AttachRow
69 NoAttach:
70 End With
71 SendEmail 'Run Send Email Macro
72 End Sub
73 Sub Email_SendAfterMtg()
74   Dim EmailLogRow, LastLogRow As Long
75   Dim MtgID As String
76   With Sheet8
77     LastLogRow = .Range("D99999").End(xlUp).Row
78     'Determine if Meeting has already been added to log
79     MtgID = Sheet1.Range("B3").Value 'Meeting ID
80     .Range("AA2:AH2,AA4:AH999").ClearContents 'Clear Previous Advanced Filter Criteria &
     results
81     .Range("AA2").Value = MtgID 'Set Meeting ID Criteria
82     .Range("AB2").Value = "Send After Meeting" 'Set Trigger Criteria
83     .Range("D5:K" & LastLogRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
     "AA1:AH2"), CopyToRange:=.Range("AA3:AH3"), unique:=True
84     If .Range("AA4").Value <> Empty Then 'Existing ID found in log
85       EmailLogRow = .Range("AH4").Value
86     Else:
87       EmailLogRow = .Range("D99999").End(xlUp).Row + 1 'First available Row
88     End If
89     .Range("D" & EmailLogRow).Value = MtgID
90     .Range("E" & EmailLogRow).Value = "Send After Meeting"
91     .Range("F" & EmailLogRow).Value = Sheet1.Range("M5").Value 'Meeting Name
92     .Range("G" & EmailLogRow).Value = Sheet1.Range("P5").Value 'Date
93     .Range("H" & EmailLogRow).Value = Sheet1.Range("M6").Value 'Start Time
94     If Sheet5.Range("G26").Value = "Days" Then
95       .Range("I" & EmailLogRow).Value = Sheet1.Range("P5").Value + Sheet5.Range("F26").
       Value + Sheet1.Range("M6").Value 'Start Date + Days + Time
96     Else:
97       .Range("I" & EmailLogRow).Value = Sheet1.Range("P5").Value + Sheet1.Range("M6").
       Value + (Sheet5.Range("F26").Value / 24) 'Start Date + Time + (Hours/24)
98     End If
99     .Range("K" & EmailLogRow).Value = "=Row()"
100 End With
101 End Sub

```

```
102 Sub Email_SendBeforeMtg()  
103 Dim EmailLogRow, LastLogRow As Long  
104 Dim MtgID As String  
105 With Sheet8  
106 LastLogRow = .Range("D99999").End(xlUp).Row  
107 'Determine if Meeting has already been added to log  
108 MtgID = Sheet1.Range("B3").Value 'Meeting ID  
109 .Range("AA2:AH2,AA4:AH999").ClearContents 'Clear Previous Advanced Filter Criteria &  
results  
110 .Range("AA2").Value = MtgID 'Set Meeting ID Criteria  
111 .Range("AB2").Value = "Send Before Meeting" 'Set Trigger Criteria  
112 .Range("D5:K" & LastLogRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(  
"AA1:AH2"), CopyToRange:=.Range("AA3:AH3"), unique:=True  
113 If .Range("AA4").Value <> Empty Then 'Existing ID found in log  
114 EmailLogRow = .Range("AH4").Value  
115 Else:  
116 EmailLogRow = Sheet8.Range("D99999").End(xlUp).Row + 1 'First available Row  
117 End If  
118 .Range("D" & EmailLogRow).Value = MtgID  
119 .Range("E" & EmailLogRow).Value = "Send Before Meeting"  
120 .Range("F" & EmailLogRow).Value = Sheet1.Range("M5").Value 'Meeting Name  
121 .Range("G" & EmailLogRow).Value = Sheet1.Range("P5").Value 'Date  
122 .Range("H" & EmailLogRow).Value = Sheet1.Range("M6").Value 'Start Time  
123 If Sheet5.Range("G27").Value = "Days" Then  
124 .Range("I" & EmailLogRow).Value = Sheet1.Range("P5").Value - Sheet5.Range("F27").  
Value + Sheet1.Range("M6").Value ' Start Date - Number of Days, + Start Time  
125 Else: 'Hours  
126 .Range("I" & EmailLogRow).Value = Sheet1.Range("P5").Value + Sheet1.Range("M6").  
Value - (Sheet5.Range("F27").Value / 24) 'Start Date + Start Time - number of  
Hours  
127 End If  
128 .Range("K" & EmailLogRow).Value = "=Row()" 'Set Database row  
129 End With  
130 End Sub
```


A

AdvancedFilter, [41](#), [42](#)
AttachEmailRow, [40](#), [41](#)
AttachRow, [40](#), [41](#)
AttendEmailRow, [40](#), [41](#)
AttendRow, [40](#), [41](#)

C

ClearContents, [40-42](#)
CopyToRange, [41](#), [42](#)
CriteriaRange, [41](#), [42](#)

E

Email_MeetingChange, [40](#)
Email_MtgStatusChange, [40](#)
Email_SelectedMeeting, [40](#)
Email_SendAfterMtg, [41](#)
Email_SendBeforeMtg, [42](#)
EmailLogRow, [41](#), [42](#)
Empty, [40-42](#)
Explicit, [40](#)

L

LastAttachRow, [40](#), [41](#)
LastAttendRow, [40](#)
LastLogRow, [41](#), [42](#)

M

MtgID, [41](#), [42](#)

N

NoAttach, [41](#)
NoAttend, [40](#), [41](#)

R

Range, [40-42](#)
Row, [40-42](#)

S

SendEmail, [40](#), [41](#)
Sheet1, [40-42](#)
Sheet5, [40-42](#)
Sheet8, [41](#), [42](#)

U

unique, [41](#), [42](#)

V

Value, [40-42](#)
vbCrLf, [41](#)

X

xlFilterCopy, [41](#), [42](#)
xlUp, [40-42](#)

```
1 Option Explicit
2
3 Sub EmailSaved_Mesg()
4     Dim i As Integer
5     Dim delay As Double
6     Dim StartTime As Double
7     With ActiveSheet.Shapes("EmailSaved" )
8         .Visible = True
9         For i = 1 To 300
10            .Fill.Transparency = i / 300
11            delay = 0.003
12            StartTime = Timer
13            Do
14                DoEvents
15            Loop While Timer - StartTime < delay
16        Next i
17        .Visible = False
18    End With
19 End Sub
```

```
20
21
22
23
24 Sub MeetingSaved_Mesg()
25     Dim i As Integer
26     Dim delay As Double
27     Dim StartTime As Double
28     With ActiveSheet.Shapes("MtgSaved" )
29         .Visible = True
30         For i = 1 To 300
31            .Fill.Transparency = i / 300
32            delay = 0.003
33            StartTime = Timer
34            Do
35                DoEvents
36            Loop While Timer - StartTime < delay
37        Next i
38        .Visible = False
39    End With
40 End Sub
```

A

ActiveSheet, [44](#)

D

delay, [44](#)

DoEvents, [44](#)

E

EmailSaved_Mesg, [44](#)

Explicit, [44](#)

F

Fill, [44](#)

I

i, [44](#)

M

MeetingSaved_Mesg, [44](#)

S

Shapes, [44](#)

StartTime, [44](#)

T

Timer, [44](#)

Transparency, [44](#)

V

Visible, [44](#)

1 Option Explicit

```
2  
3  
4 Sub MonthSel1()  
5 Sheet1.Shapes("MonthBtns").ShapeStyle = msoShapeStylePreset37  
6 Sheet1.Shapes("JanBtn").ShapeStyle = msoShapeStylePreset23  
7 Sheet1.Range("B7").Value = 1  
8 LoadMonth  
9 End Sub
```

```
10  
11 Sub MonthSel10()  
12 Sheet1.Shapes("MonthBtns").ShapeStyle = msoShapeStylePreset37  
13 Sheet1.Shapes("OctBtn").ShapeStyle = msoShapeStylePreset23  
14 Sheet1.Range("B7").Value = 10  
15 LoadMonth  
16 End Sub
```

```
17  
18 Sub MonthSel11()  
19 Sheet1.Shapes("MonthBtns").ShapeStyle = msoShapeStylePreset37  
20 Sheet1.Shapes("NovBtn").ShapeStyle = msoShapeStylePreset23  
21 Sheet1.Range("B7").Value = 11  
22 LoadMonth  
23 End Sub
```

```
24  
25 Sub MonthSel12()  
26 Sheet1.Shapes("MonthBtns").ShapeStyle = msoShapeStylePreset37  
27 Sheet1.Shapes("DecBtn").ShapeStyle = msoShapeStylePreset23  
28 Sheet1.Range("B7").Value = 12  
29 LoadMonth  
30 End Sub
```

```
31  
32 Sub MonthSel2()  
33 Sheet1.Shapes("MonthBtns").ShapeStyle = msoShapeStylePreset37  
34 Sheet1.Shapes("FebBtn").ShapeStyle = msoShapeStylePreset23  
35 Sheet1.Range("B7").Value = 2  
36 LoadMonth  
37 End Sub
```

```
38 Sub MonthSel3()  
39 Sheet1.Shapes("MonthBtns").ShapeStyle = msoShapeStylePreset37  
40 Sheet1.Shapes("MarBtn").ShapeStyle = msoShapeStylePreset23  
41 Sheet1.Range("B7").Value = 3  
42 LoadMonth  
43 End Sub
```

```
44  
45 Sub MonthSel4()  
46 Sheet1.Shapes("MonthBtns").ShapeStyle = msoShapeStylePreset37  
47 Sheet1.Shapes("AprBtn").ShapeStyle = msoShapeStylePreset23  
48 Sheet1.Range("B7").Value = 4  
49 LoadMonth  
50 End Sub
```

```
51 Sub MonthSel5()  
52 Sheet1.Shapes("MonthBtns").ShapeStyle = msoShapeStylePreset37  
53 Sheet1.Shapes("MayBtn").ShapeStyle = msoShapeStylePreset23  
54 Sheet1.Range("B7").Value = 5  
55 LoadMonth  
56 End Sub
```

```
57 Sub MonthSel6()  
58 Sheet1.Shapes("MonthBtns").ShapeStyle = msoShapeStylePreset37  
59 Sheet1.Shapes("JunBtn").ShapeStyle = msoShapeStylePreset23  
60 Sheet1.Range("B7").Value = 6  
61 LoadMonth
```

1

```
62 1 End Sub
63
64 Sub MonthSel7()
65     Sheet1.Shapes("MonthBtns").ShapeStyle = msoShapeStylePreset37
66     Sheet1.Shapes("JulBtn").ShapeStyle = msoShapeStylePreset23
67     Sheet1.Range("B7").Value = 7
68     LoadMonth
69 End Sub
70
71 Sub MonthSel8()
72     Sheet1.Shapes("MonthBtns").ShapeStyle = msoShapeStylePreset37
73     Sheet1.Shapes("AugBtn").ShapeStyle = msoShapeStylePreset23
74     Sheet1.Range("B7").Value = 8
75     LoadMonth
76 End Sub
77
78 Sub MonthSel9()
79     Sheet1.Shapes("MonthBtns").ShapeStyle = msoShapeStylePreset37
80     Sheet1.Shapes("SepBtn").ShapeStyle = msoShapeStylePreset23
81     Sheet1.Range("B7").Value = 9
82     LoadMonth
83 End Sub
```

E

Explicit, [46](#)

L

LoadMonth, [46](#), [47](#)

M

MonthSel1, [46](#)

MonthSel10, [46](#)

MonthSel11, [46](#)

MonthSel12, [46](#)

MonthSel2, [46](#)

MonthSel3, [46](#)

MonthSel4, [46](#)

MonthSel5, [46](#)

MonthSel6, [46](#)

MonthSel7, [47](#)

MonthSel8, [47](#)

MonthSel9, [47](#)

msoShapeStylePreset23, [46](#), [47](#)

msoShapeStylePreset37, [46](#), [47](#)

R

Range, [46](#), [47](#)

S

Shapes, [46](#), [47](#)

ShapeStyle, [46](#), [47](#)

Sheet1, [46](#), [47](#)

V

Value, [46](#), [47](#)

Option Explicit

```

1  Option Explicit
2
3  Sub Meeting_Load()
4      Dim MtgRow, LastAttendRow, AttendRow, LastAttendFilterRow, AttendFilterRow As Long
5      Dim SchedRow, LastAttachRow, AttachRow, LastAttachFilterRow, AttachFilterRow As Long
6      Dim MtgID As Variant
7      With Sheet1
8          If .Range("B3").Value = Empty Then
9              MsgBox "Please select a valid meeting"
10             Exit Sub
11         End If
12
13         .Range("B2").Value = True 'Set Load Meeting To True
14         MtgID = .Range("B3").Value
15         MtgRow = .Range("B4").Value
16         .Range("M5:M7,P5,P6,L11:N40,O9:P20,O23:R40").ClearContents 'Clear Existing Meeting
17         Values & Meeting ID's
18         StopCalc
19         .Range("B11:B15,B17:B20").Value = False 'Set Changed Meeting Fields to False
20         'Load Main Meeting Data
21         .Range("M5").Value = Sheet6.Range("D" & MtgRow).Value 'Meeting Name
22         .Range("P5").Value = Sheet6.Range("E" & MtgRow).Value ' Meeting Date
23         .Range("M6").Value = Sheet6.Range("F" & MtgRow).Value 'Start Time
24         .Range("M7").Value = Sheet6.Range("F" & MtgRow).Value 'Start Time
25         .Range("P6").Value = Sheet6.Range("G" & MtgRow).Value 'Duration
26         .Range("M7").Value = Sheet6.Range("H" & MtgRow).Value 'Location
27         .Range("P7").Value = Sheet6.Range("I" & MtgRow).Value 'Meeting Status
28         .Range("O9").Value = Sheet6.Range("J" & MtgRow).Value 'Agenda
29
30         'Load Meeting Attendees & Status
31         LastAttendRow = Sheet4.Range("C99999").End(xlUp).Row
32         If LastAttendRow < 4 Then GoTo NoAttend
33         Sheet4.Range("AA2:AE2,AA4:AE99999").ClearContents 'Clear Remaing Adv. Filter Criteria
34         & Results
35         Sheet4.Range("AA2").Value = MtgID
36         Sheet4.Range("C3:G" & LastAttendRow).AdvancedFilter xlFilterCopy, CriteriaRange:=
37         Sheet4.Range("AA1:AE2"), CopyToRange:=Sheet4.Range("AB3:AE3"), unique:=True
38         LastAttendFilterRow = Sheet4.Range("AC99").End(xlUp).Row
39         If LastAttendFilterRow < 3 Then GoTo NoAttend
40         For AttendFilterRow = 4 To LastAttendFilterRow
41             SchedRow = Sheet4.Range("AE" & AttendFilterRow).Value 'Schedule Row
42             .Range("L" & SchedRow & ":N" & SchedRow).Value = Sheet4.Range("AB" &
43             AttendFilterRow & ":AD" & AttendFilterRow).Value 'Copy Over Status, Name & Db Row
44         Next AttendFilterRow
45         NoAttend:
46
47         'Load Meeting Attachments
48         LastAttachRow = Sheet2.Range("C99999").End(xlUp).Row
49         If LastAttachRow < 4 Then GoTo NoAttach
50         Sheet2.Range("AA2:AE2,AA4:AE99999").ClearContents 'Clear Remaing Adv. Filter Criteria
51         & Results
52         Sheet2.Range("AA2").Value = MtgID
53         Sheet2.Range("C3:G" & LastAttachRow).AdvancedFilter xlFilterCopy, CriteriaRange:=
54         Sheet2.Range("AA1:AE2"), CopyToRange:=Sheet2.Range("AB3:AE3"), unique:=True
55         LastAttachFilterRow = Sheet2.Range("AB99").End(xlUp).Row
56         If LastAttachFilterRow < 4 Then GoTo NoAttach
57         For AttachFilterRow = 4 To LastAttachFilterRow
58             SchedRow = Sheet2.Range("AE" & AttachFilterRow).Value 'Schedule Row
59             .Range("O" & SchedRow).Value = Sheet2.Range("AB" & AttachFilterRow).Value 'File
60             Name
61             .Range("Q" & SchedRow).Value = Sheet2.Range("AD" & AttachFilterRow).Value 'Attach

```

```
1 2 3
Db Row
55 .Range("R" & SchedRow).Value = Sheet2.Range("AC" & AttachFiltRow).Value 'File
    Path
56     Next AttachFiltRow
57     NoAttach:
58
59     .Shapes("ExistMtgGrp" ).Visible = msoCTrue
60     .Shapes("NewMtgGrp" ).Visible = msoFalse
61     .Range("B2" ).Value = False 'Set Load Meeting Back To False
62     .Range("B5" ).Value = False 'Set New Meeting To False
63     End With
64
65     ResetCalc
66 End Sub
```


A

AdvancedFilter, 49
AttachFiltRow, 49, 50
AttachRow, 49
AttendFiltRow, 49
AttendRow, 49

C

ClearContents, 49
CopyToRange, 49
CriteriaRange, 49

E

Empty, 49
Explicit, 49

L

LastAttachFiltRow, 49
LastAttachRow, 49
LastAttendFiltRow, 49
LastAttendRow, 49

M

Meeting_Load, 49
MsgBox, 49
msoCTrue, 50
msoFalse, 50
MtgID, 49
MtgRow, 49

N

NoAttach, 49, 50
NoAttend, 49

R

Range, 49, 50
ResetCalc, 50
Row, 49

S

SchedRow, 49, 50
Shapes, 50
Sheet1, 49
Sheet2, 49, 50
Sheet4, 49
Sheet6, 49
StopCalc, 49

U

unique, 49

V

Value, 49, 50
Visible, 50

X

xlFilterCopy, 49
xlUp, 49

Option Explicit

Sub Mtg_AddAllByPosition()

With Sheet1

Dim FirstAttRow, LastStaffRow, LastFiltRow As Long

Dim PosType As String

.Range("B2").Value = True

PosType = .Range("M9").Value

FirstAttRow = .Range("M41").End(xlUp).Row + 1

If FirstAttRow < 11 Then FirstAttRow = 11

LastStaffRow = Sheet3.Range("C99999").End(xlUp).Row

Sheet3.Range("AA2:AE2,AA4:AE9999").ClearContents 'Clear Previous Criteria & Filters

Sheet3.Range("AC2").Value = PosType

Sheet3.Range("C2:G" & LastStaffRow).AdvancedFilter xlFilterCopy, CriteriaRange:=Sheet3

.Range("AA1:AE2"), CopyToRange:=Sheet3.Range("AA3:AE3"), unique:=True

LastFiltRow = Sheet3.Range("AA9999").End(xlUp).Row

If LastFiltRow < 4 Then GoTo NoStaff

.Range("M" & FirstAttRow & ":M" & FirstAttRow + LastFiltRow - 4).Value = Sheet3.Range

("AA4:AA" & LastFiltRow).Value 'Copy Over Filtered Staff Names

.Range("L" & FirstAttRow & ":L" & FirstAttRow + LastFiltRow - 4).Value = "Tentative"

NoStaff:

.Range("B2").Value = False

End With

End Sub

Sub Mtg_DeleteAttachment()

Dim AttachRow, ActRow As Long

With Sheet1

ActRow = ActiveCell.Row

If .Range("Q" & ActRow).Value <> Empty Then

AttachRow = .Range("Q" & ActiveCell.Row).Value 'Attendee Database Row

Sheet2.Range(AttachRow & ":" & AttachRow).EntireRow.Delete

End If

.Range("O" & ActRow & ":R" & ActRow).ClearContents

End With

End Sub

Sub Mtg_DeleteAttendee()

Dim AttendRow, ActRow As Long

With Sheet1

ActRow = ActiveCell.Row

If .Range("N" & ActRow).Value <> Empty Then

AttendRow = .Range("N" & ActiveCell.Row).Value 'Attendee Database Row

Sheet4.Range(AttendRow & ":" & AttendRow).EntireRow.Delete

End If

.Range("L" & ActRow & ":N" & ActRow).ClearContents

End With

End Sub

A

ActiveCell, [52](#)
ActRow, [52](#)
AdvancedFilter, [52](#)
AttachRow, [52](#)
AttendRow, [52](#)

C

ClearContents, [52](#)
CopyToRange, [52](#)
CriteriaRange, [52](#)

D

Delete, [52](#)

E

Empty, [52](#)
EntireRow, [52](#)
Explicit, [52](#)

F

FirstAttRow, [52](#)

L

LastFiltRow, [52](#)
LastStaffRow, [52](#)

M

Mtg_AddAllByPosition, [52](#)
Mtg_DeleteAttachment, [52](#)
Mtg_DeleteAttendee, [52](#)

N

NoStaff, [52](#)

P

PosType, [52](#)

R

Range, [52](#)
Row, [52](#)

S

Sheet1, [52](#)
Sheet2, [52](#)
Sheet3, [52](#)
Sheet4, [52](#)

U

unique, [52](#)

V

Value, [52](#)

X

xlFilterCopy, [52](#)
xlUp, [52](#)

Option Explicit

```
1
2
3 Sub Meeting_Delete()
4   Dim MtgRow As Long
5   If MsgBox("Are you sure you want to delete this meeting?" , vbYesNo, "Delete Meeting" ) =
6     vbNo Then Exit Sub
7   If Sheet1.Range("B4" ).Value = Empty Then Exit Sub
8   MtgRow = Sheet1.Range("B4" ).Value 'Meeting Row
9   Sheet6.Range(MtgRow & ":" & MtgRow).EntireRow.Delete
10  Meeting_New 'Clear Fields and set to New Meeting
11  LoadMonth 'Reload Month View
12 End Sub
```

```
13 Sub Meeting_New()
14   With Sheet1
15     .Range("B5" ).Value = True 'Set New Meeting to True
16     .Range("B3,M5:M7,P5,P6,L11:N40,O9:P20,O23:R40" ).ClearContents 'Clear Existing Meeting
17     Values
18     .Range("B11:B15,B17:B20" ).Value = False 'Set Changed Meeting Fields to False
19     .Shapes("NewMtgGrp" ).Visible = msoCTrue
20     .Shapes("ExistMtgGrp" ).Visible = msoFalse
21     .Range("P7" ).Value = "Pending" 'Set Default Meeting Status
22     .Range("M5" ).Select
23   End With
24 End Sub
```

```
25
26
27 Sub OutlookTest()
28   Dim OutApp, myItem As Object
29   Dim myRequiredAttendee, myOptionalAttendee, myResourceAttendee As Outlook.Recipient
30   Set OutApp = CreateObject("Outlook.Application" )
31   Set myItem = OutApp.CreateItem(olAppointmentItem)
32   myItem.MeetingStatus = olMeeting
33   myItem.Subject = "Strategy Meeting"
34   myItem.Location = "Conf Rm All Stars"
35   myItem.Start = #8/4/2018 1:30:00 PM#
36   myItem.Duration = 90
37   Set myRequiredAttendee = myItem.Recipients.Add("Nate Sun" )
38   myRequiredAttendee.Type = olRequired
39   Set myOptionalAttendee = myItem.Recipients.Add("Kevin Kennedy" )
40   myOptionalAttendee.Type = olOptional
41   ' Set myResourceAttendee = myItem.Recipients.Add("Conf Rm All Stars")
42   ' myResourceAttendee.Type = olResource
43   myItem.Display
44   ' myItem.Send
45 End Sub
46
47
```

A

Add, [54](#)

C

ClearContents, [54](#)

CreateItem, [54](#)

CreateObject, [54](#)

D

Delete, [54](#)

Display, [54](#)

Duration, [54](#)

E

Empty, [54](#)

EntireRow, [54](#)

Explicit, [54](#)

L

LoadMonth, [54](#)

Location, [54](#)

M

Meeting_Delete, [54](#)

Meeting_New, [54](#)

MeetingStatus, [54](#)

MsgBox, [54](#)

msoCTrue, [54](#)

msoFalse, [54](#)

MtgRow, [54](#)

myItem, [54](#)

myOptionalAttendee, [54](#)

myRequiredAttendee, [54](#)

myResourceAttendee, [54](#)

O

olAppointmentItem, [54](#)

olMeeting, [54](#)

olOptional, [54](#)

olRequired, [54](#)

OutApp, [54](#)

Outlook, [54](#)

OutlookTest, [54](#)

P

PM, [54](#)

R

Range, [54](#)

Recipient, [54](#)

Recipients, [54](#)

S

Shapes, [54](#)

Sheet1, [54](#)

Sheet6, [54](#)

Start, [54](#)

Subject, [54](#)

V

Value, [54](#)

vbNo, [54](#)

vbYesNo, [54](#)

Visible, [54](#)

```

1  Option Explicit
2
3  Sub SaveUpdate_Meeting()
4      Dim MtgRow, AttendRow, LastAttendRow, AttendDbRow As Long
5      Dim AttachRow, LastAttachRow, AttachDbRow As Long
6      Dim MtgID As Variant
7      Dim AttendStatusRng, FoundAttendStatusRng As Range
8      Dim Status As String
9      Set AttendStatusRng = Sheet1.Range("AttendStatus" )
10     With Sheet1
11         If .Range("B5" ).Value = True Then 'New Meeting
12             MtgRow = Sheet6.Range("C99999" ).End(xlUp).Row + 1 'First Available Row
13             MtgID = .Range("B6" ).Value
14             .Range("B3" ).Value = MtgID
15             If Sheet5.Range("D21" ).Value = True Then 'If New Meeting Email is Active
16                 Sheet5.Range("B30" ).Value = 21 'New Meeting Trigger Email Row
17                 Email_SelectedMeeting 'Email New Meeting if New Meeting is Active
18             End If
19         Else: 'Existing Meeting
20             If .Range("B4" ).Value = Empty Then
21                 MsgBox "Please select a correct meeting to edit"
22                 Exit Sub
23             End If
24             MtgRow = .Range("B4" ).Value
25             MtgID = .Range("B3" ).Value
26             If Sheet5.Range("D22" ).Value = True Then
27                 Email_MeetingChange 'Email if Meeting Change is Active
28                 'GoTo SkipStatusChange
29             End If
30             If Sheet5.Range("D25" ).Value = True Then Email_MtgStatusChange 'Email if Meeting
31             Status Change is Active
32             SkipStatusChange:
33             'Check for New Attendees and run macro if both are true
34             If Sheet5.Range("D23" ).Value = True Then
35                 Sheet5.Range("B30" ).Value = 23 'Set Email Automation Row
36                 Email_SelectedMeeting
37             End If
38         End If
39     End With
40     StopCalc
41     Sheet6.Range("C" & MtgRow).Value = MtgID 'Meeting ID
42     Sheet6.Range("D" & MtgRow).Value = .Range("M5" ).Value 'Meeting Name
43     Sheet6.Range("E" & MtgRow).Value = .Range("P5" ).Value ' Meeting Date
44     Sheet6.Range("F" & MtgRow).Value = .Range("M6" ).Value 'Start Time
45     Sheet6.Range("G" & MtgRow).Value = .Range("P6" ).Value 'Duration
46     Sheet6.Range("H" & MtgRow).Value = .Range("M7" ).Value 'Location
47     Sheet6.Range("I" & MtgRow).Value = .Range("P7" ).Value 'Status
48     Sheet6.Range("J" & MtgRow).Value = .Range("O9" ).Value 'Agenda
49
50     'Add/Update Attendees
51     LastAttendRow = .Range("M41" ).End(xlUp).Row
52     If LastAttendRow < 11 Then GoTo NoAttendees
53     For AttendRow = 11 To LastAttendRow
54         If .Range("M" & AttendRow).Value <> Empty Then 'Check for actual Attendee
55             If .Range("N" & AttendRow).Value <> Empty Then
56                 AttendDbRow = .Range("N" & AttendRow).Value
57             Else:
58                 AttendDbRow = Sheet4.Range("C99999" ).End(xlUp).Row + 1
59                 .Range("N" & AttendRow).Value = AttendDbRow 'Add in Database Row
60             End If

```

1 2 3 4

```

1 2 3 4
61 Sheet4.Range("C" & AttendDbRow).Value = MtgID 'Meeting ID
62 Sheet4.Range("D" & AttendDbRow).Value = .Range("M" & AttendRow).Value 'Name
63 Status = .Range("L" & AttendRow).Value
64 'Check For existing Status, and if different AND Attendee Status Email Is
Active, then send email
65 If Sheet4.Range("E" & AttendDbRow).Value <> Empty And Sheet4.Range("E" &
AttendDbRow).Value <> Status And Sheet5.Range("D25" ).Value = True Then
66 Set FoundAttendStatusRng = AttendStatusRng.Find(Status, , xlValues, xlWhole)
67 If Not FoundAttendStatusRng Is Nothing Then 'Find Status Row
68 If Sheet5.Cells(24, FoundAttendStatusRng.Row - 90 + 12).Value = True Then
69 Sheet5.Range("B30" ).Value = 24 'Set Email Automation Row
70 Sheet5.Range("A44" ).Value = .Range("M" & AttendRow).Value
71 Email_SelectedMeeting 'Send Email Update
72 End If
73 End If
74 End If
75
76 Sheet4.Range("E" & AttendDbRow).Value = .Range("L" & AttendRow).Value 'Status
77 Sheet4.Range("F" & AttendDbRow).Value = AttendRow 'Attendachment Schedule Row
78 Sheet4.Range("G" & AttendDbRow).Value = "=Row()" 'Database Row
79 Sheet4.Range("H" & AttendDbRow).Value = "=C" & AttendDbRow & "&" & Chr(34) &
"_" & Chr(34) & "&" & "D" & AttendDbRow
80 End If
81 Next AttendRow
82 NoAttendees:
83
84 'Add/Update Attendachments
85 LastAttachRow = .Range("O41" ).End(xlUp).Row
86 If LastAttachRow < 23 Then GoTo NoAttachments
87 For AttachRow = 23 To LastAttachRow
88 If .Range("O" & AttachRow).Value <> Empty Then 'Check for actual Attachment
89 If .Range("Q" & AttachRow).Value <> Empty Then
90 AttachDbRow = .Range("Q" & AttachRow).Value
91 Else:
92 AttachDbRow = Sheet2.Range("C99999" ).End(xlUp).Row + 1
93 .Range("Q" & AttachRow).Value = AttachDbRow 'Add in Database Row
94 End If
95 Sheet2.Range("C" & AttachDbRow).Value = MtgID 'Meeting ID
96 Sheet2.Range("D" & AttachDbRow).Value = .Range("O" & AttachRow).Value
'Filename
97 Sheet2.Range("E" & AttachDbRow).Value = .Range("R" & AttachRow).Value 'Full
File Path
98 Sheet2.Range("F" & AttachDbRow).Value = AttachRow 'Attachachment Schedule Row
99 Sheet2.Range("G" & AttachDbRow).Value = "=Row()" 'Database Row
100 End If
101 Next AttachRow
102 NoAttachments:
103 .Shapes("NewMtgGrp" ).Visible = msoFalse
104 .Shapes("ExistMtgGrp" ).Visible = msoCTrue
105 .Range("B5" ).Value = False 'Set New Meeting To False
106 .Range("B11:B15,B17:B20" ).Value = False 'set Meting Fields & Status Change To False
107 End With
108 'Check for Send After & Before Meeting Emails
109 If Sheet5.Range("D26" ).Value = True Then Email_SendAfterMtg
110 If Sheet5.Range("D27" ).Value = True Then Email_SendBeforeMtg
111 Email_ScheduleSend
112 ResetCalc
113 LoadMonth 'Refresh Schedule
114 MeetingSaved_Mesg
115 End Sub

```

A

AttachDbRow, [56](#), [57](#)
AttachRow, [56](#), [57](#)
AttendDbRow, [56](#), [57](#)
AttendRow, [56](#), [57](#)
AttendStatusRng, [56](#), [57](#)

C

Cells, [57](#)
Chr, [57](#)

E

Email_MeetingChange, [56](#)
Email_MtgStatusChange, [56](#)
Email_ScheduleSend, [57](#)
Email_SelectedMeeting, [56](#), [57](#)
Email_SendAfterMtg, [57](#)
Email_SendBeforeMtg, [57](#)
Empty, [56](#), [57](#)
Explicit, [56](#)

F

Find, [57](#)
FoundAttendStatusRng, [56](#), [57](#)

L

LastAttachRow, [56](#), [57](#)
LastAttendRow, [56](#)
LoadMonth, [57](#)

M

MeetingSaved_Mesg, [57](#)
MsgBox, [56](#)
msoCTrue, [57](#)
msoFalse, [57](#)
MtgID, [56](#), [57](#)
MtgRow, [56](#)

N

NoAttachments, [57](#)
NoAttendees, [56](#), [57](#)

R

Range, [56](#), [57](#)
ResetCalc, [57](#)
Row, [56](#), [57](#)

S

SaveUpdate_Meeting, [56](#)
Shapes, [57](#)
Sheet1, [56](#)
Sheet2, [57](#)
Sheet4, [56](#), [57](#)
Sheet5, [56](#), [57](#)
Sheet6, [56](#)
SkipStatusChange, [56](#)
Status, [56](#), [57](#)
StopCalc, [56](#)

V

Value, [56](#), [57](#)
Visible, [57](#)

X

xlUp, [56](#), [57](#)
xlValues, [57](#)
xlWhole, [57](#)


```
1 Dim SelCell As Range
2 Dim DayName As String
3
4 .....Color Calendar Background'.....
5 Sub CalCol1()
6 With ActiveSheet.Shapes.Range(Array("CalBack" , "Settings" )).Select
7 With Selection.ShapeRange.Fill
8 .ForeColor.RGB = RGB(234, 234, 234)
9 End With
10 ActiveSheet.Range(Sheets("CalPopUp" ).Range("A7" ).Value).Select
11 End With
12 End Sub
13 Sub CalCol2()
14 With ActiveSheet.Shapes.Range(Array("CalBack" , "Settings" )).Select
15 With Selection.ShapeRange.Fill
16 .ForeColor.RGB = RGB(197, 190, 151)
17 End With
18 ActiveSheet.Range(Sheets("CalPopUp" ).Range("A7" ).Value).Select
19 End With
20 End Sub
21 Sub CalCol3()
22 With ActiveSheet.Shapes.Range(Array("CalBack" , "Settings" )).Select
23 With Selection.ShapeRange.Fill
24 .ForeColor.RGB = RGB(141, 180, 227)
25 End With
26 ActiveSheet.Range(Sheets("CalPopUp" ).Range("A7" ).Value).Select
27 End With
28 End Sub
29 Sub CalCol4()
30 With ActiveSheet.Shapes.Range(Array("CalBack" , "Settings" )).Select
31 With Selection.ShapeRange.Fill
32 .ForeColor.RGB = RGB(184, 204, 228)
33 End With
34 ActiveSheet.Range(Sheets("CalPopUp" ).Range("A7" ).Value).Select
35 End With
36 End Sub
37 Sub CalCol5()
38 With ActiveSheet.Shapes.Range(Array("CalBack" , "Settings" )).Select
39 With Selection.ShapeRange.Fill
40 .ForeColor.RGB = RGB(230, 185, 184)
41 End With
42 ActiveSheet.Range(Sheets("CalPopUp" ).Range("A7" ).Value).Select
43 End With
44 End Sub
45 Sub CalCol6()
46 With ActiveSheet.Shapes.Range(Array("CalBack" , "Settings" )).Select
47 With Selection.ShapeRange.Fill
48 .ForeColor.RGB = RGB(215, 228, 188)
49 End With
50 ActiveSheet.Range(Sheets("CalPopUp" ).Range("A7" ).Value).Select
51 End With
52 End Sub
53 Sub CalCol7()
54 With ActiveSheet.Shapes.Range(Array("CalBack" , "Settings" )).Select
55 With Selection.ShapeRange.Fill
56 .ForeColor.RGB = RGB(204, 192, 218)
57 End With
58 ActiveSheet.Range(Sheets("CalPopUp" ).Range("A7" ).Value).Select
59 End With
60 End Sub
61 Sub CalCol8()
```

1

```

1
62 With ActiveSheet.Shapes.Range(Array("CalBack" , "Settings" )).Select
63 With Selection.ShapeRange.Fill
64 .ForeColor.RGB = RGB(182, 221, 232)
65 End With
66 ActiveSheet.Range(Sheets("CalPopUp" ).Range("A7" ).Value).Select
67 End With
68 End Sub
69 Sub CalCol9()
70 With ActiveSheet.Shapes.Range(Array("CalBack" , "Settings" )).Select
71 With Selection.ShapeRange.Fill
72 .ForeColor.RGB = RGB(252, 213, 180)
73 End With
74 ActiveSheet.Range(Sheets("CalPopUp" ).Range("A7" ).Value).Select
75 End With
76 End Sub
77 Sub CalendarHide()
78 Dim DayNum As Long
79 'Hide Calendar, Reset Day Colors
80 On Error GoTo NoCal
81 ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
82 Sheets("CalPopUp" ).Range("A7" ).Value = ""
83 If Sheets("CalPopUp" ).Range("A20" ).Value <> Empty Then
84 For DayNum = 1 To 42
85 DayName = DayNum & "Day"
86 With ActiveSheet.Shapes(DayName)
87 .Fill.ForeColor.RGB = RGB(255, 255, 255)
88 .TextFrame2.TextRange.Font.Bold = msoFalse
89 End With
90 Next DayNum
91 End If
92 Exit Sub
93 NoCal: 'If calendar has been removed by accident, paste in backup calendar from CalPopUp
Sheet
94 End Sub
95
96 Sub CalendarShow()
97 With ActiveSheet
98 Set SelCell = Selection
99 'Check if active cell is a valid date
100 If IsDate(SelCell.Value) = True Then
101 Sheets("CalPopUp" ).Range("A1" ).Value = SelCell.Value
102 Else: 'If No Date or incorrect Date user current date
103 Sheets("CalPopUp" ).Range("A1" ).Value = "=Today()"
104 End If
105 Sheets("CalPopUp" ).Range("A3" ).Value = Month(Sheets("CalPopUp" ).Range("A1" ).Value)
'Set Month
106 Sheets("CalPopUp" ).Range("A2" ).Value = Year(Sheets("CalPopUp" ).Range("A1" ).Value)
'Set Year
107 DayName = Sheets("CalPopUp" ).Range("A20" ).Value & "Day"
108 UngroupCal
109 If InStr(.Shapes("NextTri" ).OnAction, "!" ) <> 0 Or InStr(.Shapes("1Day" ).
DrawingObject.Formula, "]" ) <> 0 Then 'Run Workbook Link Remover and Cell Link
Replacement
110 MacroLinkRemover
111 CalFormulaReplacement
112 End If
113 GroupCal
114 On Error GoTo NoCal
115 With ActiveSheet.Shapes(DayName)
116 .Fill.ForeColor.RGB = RGB(252, 213, 180)
117 .TextFrame2.TextRange.Font.Bold = msoTrue

```

1 2 3

```

1 2 3
118 End With
119 On Error GoTo NoCal
120 .Shapes("Calendar").Visible = msoCTrue
121 .Shapes.Range(Array("Settings", "CalCol1", "CalCol2", "CalCol3", "CalCol4",
122 "CalCol5", "CalCol6", "CalCol7", "CalCol8", "CalCol9")).Visible = False
123 .Shapes("Calendar").Left = SelCell.Left
124 .Shapes("Calendar").Top = SelCell.Offset(1, 0).Top
125 If Sheets("CalPopUp").Range("A6").Value > 0 Then
126 .Shapes.Range(Array("36Day", "37Day", "38Day", "39Day", "40Day", "41Day",
127 "42Day")).Visible = True
128 Else:
129 .Shapes.Range(Array("36Day", "37Day", "38Day", "39Day", "40Day", "41Day",
130 "42Day")).Visible = False
131 End If
132 Sheets("CalPopUp").Range("A7").Value = SelCell.Address
133 ActiveCell.Select
134 End With
135 Exit Sub
136 NoCal:
137 MsgBox "The Pop-up Calendar does not exist on this worksheet. Please copy the calendar
138 over from another sheet and paste into this sheet"
139 End Sub
140 Sub CalFormulaReplacement()
141 With ActiveSheet
142 Dim DayNum, ColNum, RowNum As Long
143 Dim Shp As Shape
144 ColNum = 2
145 RowNum = 1
146 For DayNum = 1 To 42
147 .Shapes(DayNum & "Day").DrawingObject.Formula = "=CalPopUp!" & .Cells(RowNum,
148 ColNum).Address
149 ColNum = ColNum + 1
150 If ColNum = 9 Then
151 ColNum = 2
152 RowNum = RowNum + 1
153 End If
154 Next DayNum
155 .Shapes("MonthYear").DrawingObject.Formula = "=CalPopUp!$A$5"
156 End With
157 End Sub
158 Sub CheckForSheet()
159 'Checks for existence of Calendar Pop-up Worksheet
160 Dim ws As Worksheet
161 On Error GoTo CreateWS
162 Set ws = ActiveWorkbook.Sheets("CalPopUp")
163 Exit Sub
164 CreateWS:
165 CreateCalSht
166 End Sub
167 'Create Calendar Sheet on First Run of Calendar
168 Sub CreateCalSht()
169 Dim ColCnt, RowCnt, DayCnt, CalCol As Long
170 Dim ws, ActSht As Worksheet
171 Set ActSht = ActiveSheet
172 'On Error GoTo NoCal
173 ActiveSheet.Shapes("Calendar").Copy
174 Set ws = ThisWorkbook.Sheets.Add(After:=ThisWorkbook.Sheets(ThisWorkbook.Sheets.Count))
175 ws.Name = "CalPopUp"
176 ActSht.Activate

```

1

```

174
175 'Reassign Shape Links & Macros
176 With ActiveSheet
177     UnGroupCal
178     '.Unprotect
179     .Shapes("NextRec" ).OnAction = "" & ActiveWorkbook.Name & "!NextMonth"
180     .Shapes("NextTri" ).OnAction = "" & ActiveWorkbook.Name & "!NextMonth"
181     .Shapes("PrevRec" ).OnAction = "" & ActiveWorkbook.Name & "!PrevMonth"
182     .Shapes("PrevTri" ).OnAction = "" & ActiveWorkbook.Name & "!PrevMonth"
183     .Shapes("SetBtn" ).OnAction = "" & ActiveWorkbook.Name & "!ShowSettings"
184     .Shapes("MonthYear" ).DrawingObject.Formula = "=CalPopUp!A5"
185     DayCnt = 1
186     For RowCnt = 1 To 6
187         For ColCnt = 2 To 8
188             .Shapes(DayCnt & "Day" ).DrawingObject.Formula = "=CalPopUp!" & .Cells(RowCnt,
189             ColCnt).Address 'Assigned Linked Cell
190             .Shapes(DayCnt & "Day" ).OnAction = "" & ActiveWorkbook.Name & "!DayBtn" &
191             DayCnt 'Assign Macro
192             DayCnt = DayCnt + 1
193         Next ColCnt
194     Next RowCnt
195
196 'Assign Color Macros
197 For CalCol = 1 To 9
198     .Shapes("CalCol" & CalCol).OnAction = "" & ActiveWorkbook.Name & "!CalCol" &
199     CalCol
200 Next CalCol
201 End With
202
203 With Sheets("CalPopUp" )
204     .Paste
205     .Visible = xlSheetHidden
206
207 'Add in Formulas and Details
208 .Range("A4" ).Value = "=INDEX(CalMonths,A3,)"
209 .Range("A5" ).Value = "=A4&" & Chr(34) & " " & Chr(34) & "&CalYear"
210 .Range("A6" ).Value = "=SUM(B6:H6)"
211 .Range("A8" ).Value = "January"
212 .Range("A8" ).AutoFill Destination:=.Range("A8:A19" ), Type:=xlFillDefault
213 .Range("A20" ).Value =
214 "=IFERROR(INDIRECT(ADDRESS(SUMPRODUCT((B1:H6=A1)*ROW(B1:H6))+6,SUMPRODUCT((B1:H6=A1)*COL
215 UMN(B1:H6)),1,1)),)" & Chr(34) & Chr(34) & ""
216 'Set Defined Names
217 ActiveWorkbook.Names.Add Name:="CalMonths" , RefersTo:="=CalPopUp!$A$8:$A$19"
218 ActiveWorkbook.Names.Add Name:="CalYear" , RefersTo:="=CalPopUp!$A$2"
219
220 'Add in Calendar Formulas
221 .Range("B1" ).Value =
222 "=IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=1,DATE(CalYear,MATCH($A$4,CalMonth
223 s,0),1)," & Chr(34) & Chr(34) & ""
224 .Range("C1" ).Value = "=IF(B1<>" & Chr(34) & Chr(34) &
225 ",B1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=2,DATE(CalYear,MATCH($A$4,Cal
226 Months,0),1)," & Chr(34) & Chr(34) & ""
227 .Range("D1" ).Value = "=IF(C1<>" & Chr(34) & Chr(34) &
228 ",C1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=3,DATE(CalYear,MATCH($A$4,Cal
229 Months,0),1)," & Chr(34) & Chr(34) & ""
230 .Range("E1" ).Value = "=IF(D1<>" & Chr(34) & Chr(34) &
231 ",D1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=4,DATE(CalYear,MATCH($A$4,Cal
232 Months,0),1)," & Chr(34) & Chr(34) & ""
233 .Range("F1" ).Value = "=IF(E1<>" & Chr(34) & Chr(34) &
234 ",E1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=5,DATE(CalYear,MATCH($A$4,Cal

```

1 2

```

1 2
221 .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$4,Cal
Months,0),1)," & Chr(34) & Chr(34) & ")"
222 .Range("H1").Value = "=IF(G1<>" & Chr(34) & Chr(34) & "
    ",G1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=7,DATE(CalYear,MATCH($A$4,Cal
Months,0),1)," & Chr(34) & Chr(34) & ")"
223 .Range("B2").Value = "=H1+1"
224 .Range("C2").Value = "=B2+1"
225 .Range("C2").AutoFill Destination:=.Range("C2:H2"), Type:=xlFillDefault
226 .Range("B2:H2").AutoFill Destination:=.Range("B2:H4"), Type:=xlFillDefault
227 .Range("B5").Value = "=IF(OR(H4=" & Chr(34) & Chr(34) & ",MONTH(H4+1)<>$A$3)," & Chr
(34) & Chr(34) & ",H4+1)"
228 .Range("C5").Value = "=IFERROR(IF(MONTH(B5+1)<>$A$3," & Chr(34) & Chr(34) & "
    ",B5+1)," & Chr(34) & Chr(34) & ")"
229 .Range("B6").Value = "=IFERROR(IF(OR(H5=" & Chr(34) & Chr(34) & "
    ",MONTH(H5+1)<>$A$3)," & Chr(34) & Chr(34) & ",H5+1)," & Chr(34) & Chr(34) & ")"
230 .Range("C6").Value = "=IFERROR(F(MONTH(I5+1)<>$A$3," & Chr(34) & Chr(34) & ",I5+1),"
    & Chr(34) & Chr(34) & ")"
231 .Range("C5:C6").AutoFill Destination:=.Range("C5:H6"), Type:=xlFillDefault
232
233 'Set format to Single Day
234 .Range("B1:H6").NumberFormat = "d"
235
236 'Add in relative Day #'s
237 .Range("B7").Value = "1"
238 .Range("C7").Value = "2"
239 .Range("B8").Value = "8"
240 .Range("C8").Value = "9"
241 .Range("B7:C8").AutoFill Destination:=.Range("B7:H8"), Type:=xlFillDefault
242 .Range("B7:H8").AutoFill Destination:=.Range("B7:H12"), Type:=xlFillDefault
243 GroupCal
244 End With
245 Exit Sub
246 NoCal:
247 MsgBox "The Pop-up Calendar does not exist on this worksheet. Please copy the calendar
    over from another sheet and paste into this sheet"
248 End Sub
249
250 .....
251 ''''Individual Day Selection Buttons
252 .....
253
254 Sub DayBtn1()
255 With Sheets("CalPopUp")
256 If .Range("B1").Value <> Empty And .Range("A7").Value <> Empty Then
257     ActiveSheet.Range(.Range("A7").Value).Value = .Range("B1").Value
258 End If
259 ActiveSheet.Shapes("Calendar").Visible = msoFalse
260 ActiveCell.Offset(0, 1).Select
261 End With
262 End Sub
263 Sub DayBtn10()
264 With Sheets("CalPopUp")
265 If .Range("D2").Value <> Empty And .Range("A7").Value <> Empty Then
266     ActiveSheet.Range(.Range("A7").Value).Value = .Range("D2").Value
267 End If
268 ActiveSheet.Shapes("Calendar").Visible = msoFalse
269 ActiveCell.Offset(0, 1).Select
270 End With
271 End Sub

```

```
272 Sub DayBtn11()  
273 With Sheets("CalPopUp" )  
274 If .Range("E2" ).Value <> Empty And .Range("A7" ).Value <> Empty Then  
275     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("E2" ).Value  
276 End If  
277     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse  
278     ActiveCell.Offset(0, 1).Select  
279 End With  
280 End Sub  
281 Sub DayBtn12()  
282 With Sheets("CalPopUp" )  
283 If .Range("F2" ).Value <> Empty And .Range("A7" ).Value <> Empty Then  
284     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("F2" ).Value  
285 End If  
286     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse  
287     ActiveCell.Offset(0, 1).Select  
288 End With  
289 End Sub  
290 Sub DayBtn13()  
291 With Sheets("CalPopUp" )  
292 If .Range("G2" ).Value <> Empty And .Range("A7" ).Value <> Empty Then  
293     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("G2" ).Value  
294 End If  
295     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse  
296     ActiveCell.Offset(0, 1).Select  
297 End With  
298 End Sub  
299 Sub DayBtn14()  
300 With Sheets("CalPopUp" )  
301 If .Range("H2" ).Value <> Empty And .Range("A7" ).Value <> Empty Then  
302     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("H2" ).Value  
303 End If  
304     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse  
305     ActiveCell.Offset(0, 1).Select  
306 End With  
307 End Sub  
308 Sub DayBtn15()  
309 With Sheets("CalPopUp" )  
310 If .Range("B3" ).Value <> Empty And .Range("A7" ).Value <> Empty Then  
311     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("B3" ).Value  
312 End If  
313     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse  
314     ActiveCell.Offset(0, 1).Select  
315 End With  
316 End Sub  
317 Sub DayBtn16()  
318 With Sheets("CalPopUp" )  
319 If .Range("C3" ).Value <> Empty And .Range("A7" ).Value <> Empty Then  
320     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("C3" ).Value  
321 End If  
322     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse  
323     ActiveCell.Offset(0, 1).Select  
324 End With  
325 End Sub  
326 Sub DayBtn17()  
327 With Sheets("CalPopUp" )  
328 If .Range("D3" ).Value <> Empty And .Range("A7" ).Value <> Empty Then  
329     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("D3" ).Value  
330 End If  
331     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse  
332     ActiveCell.Offset(0, 1).Select
```

```
1 2
333 End With
334 End Sub
335 Sub DayBtn18()
336 With Sheets("CalPopUp" )
337 If .Range("E3" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
338     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("E3" ).Value
339 End If
340 ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
341 ActiveCell.Offset(0, 1).Select
342 End With
343 End Sub
344 Sub DayBtn19()
345 With Sheets("CalPopUp" )
346 If .Range("F3" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
347     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("F3" ).Value
348 End If
349 ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
350 ActiveCell.Offset(0, 1).Select
351 End With
352 End Sub
353 Sub DayBtn2()
354 With Sheets("CalPopUp" )
355 If .Range("C1" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
356     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("C1" ).Value
357 End If
358 ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
359 ActiveCell.Offset(0, 1).Select
360 End With
361 End Sub
362 Sub DayBtn20()
363 With Sheets("CalPopUp" )
364 If .Range("G3" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
365     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("G3" ).Value
366 End If
367 ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
368 ActiveCell.Offset(0, 1).Select
369 End With
370 End Sub
371 Sub DayBtn21()
372 With Sheets("CalPopUp" )
373 If .Range("H3" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
374     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("H3" ).Value
375 End If
376 ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
377 ActiveCell.Offset(0, 1).Select
378 End With
379 End Sub
380 Sub DayBtn22()
381 With Sheets("CalPopUp" )
382 If .Range("B4" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
383     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("B4" ).Value
384 End If
385 ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
386 ActiveCell.Offset(0, 1).Select
387 End With
388 End Sub
389 Sub DayBtn23()
390 With Sheets("CalPopUp" )
391 If .Range("C4" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
392     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("C4" ).Value
393 End If
```

1 2


```
1 2
394 ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
395 ActiveCell.Offset(0, 1).Select
396 End With
397 End Sub
398 Sub DayBtn24()
399 With Sheets("CalPopUp" )
400 If .Range("D4" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
401 ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("D4" ).Value
402 End If
403 ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
404 ActiveCell.Offset(0, 1).Select
405 End With
406 End Sub
407 Sub DayBtn25()
408 With Sheets("CalPopUp" )
409 If .Range("E4" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
410 ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("E4" ).Value
411 End If
412 ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
413 ActiveCell.Offset(0, 1).Select
414 End With
415 End Sub
416 Sub DayBtn26()
417 With Sheets("CalPopUp" )
418 If .Range("F4" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
419 ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("F4" ).Value
420 End If
421 ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
422 ActiveCell.Offset(0, 1).Select
423 End With
424 End Sub
425 Sub DayBtn27()
426 With Sheets("CalPopUp" )
427 If .Range("G4" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
428 ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("G4" ).Value
429 End If
430 ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
431 ActiveCell.Offset(0, 1).Select
432 End With
433 End Sub
434 Sub DayBtn28()
435 With Sheets("CalPopUp" )
436 If .Range("H4" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
437 ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("H4" ).Value
438 End If
439 ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
440 ActiveCell.Offset(0, 1).Select
441 End With
442 End Sub
443 Sub DayBtn29()
444 With Sheets("CalPopUp" )
445 If .Range("B5" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
446 ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("B5" ).Value
447 End If
448 ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
449 ActiveCell.Offset(0, 1).Select
450 End With
451 End Sub
452 Sub DayBtn3()
453 With Sheets("CalPopUp" )
454 If .Range("D1" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
```

1 2 3


```
1 2 3
455     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("D1" ).Value
456     End If
457     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
458     ActiveCell.Offset(0, 1).Select
459     End With
460 End Sub
461 Sub DayBtn30()
462     With Sheets("CalPopUp" )
463     If .Range("C5" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
464         ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("C5" ).Value
465     End If
466     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
467     ActiveCell.Offset(0, 1).Select
468     End With
469 End Sub
470 Sub DayBtn31()
471     With Sheets("CalPopUp" )
472     If .Range("D5" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
473         ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("D5" ).Value
474     End If
475     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
476     ActiveCell.Offset(0, 1).Select
477     End With
478 End Sub
479 Sub DayBtn32()
480     With Sheets("CalPopUp" )
481     If .Range("E5" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
482         ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("E5" ).Value
483     End If
484     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
485     ActiveCell.Offset(0, 1).Select
486     End With
487 End Sub
488 Sub DayBtn33()
489     With Sheets("CalPopUp" )
490     If .Range("F5" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
491         ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("F5" ).Value
492     End If
493     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
494     ActiveCell.Offset(0, 1).Select
495     End With
496 End Sub
497 Sub DayBtn34()
498     With Sheets("CalPopUp" )
499     If .Range("G5" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
500         ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("G5" ).Value
501     End If
502     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
503     ActiveCell.Offset(0, 1).Select
504     End With
505 End Sub
506 Sub DayBtn35()
507     With Sheets("CalPopUp" )
508     If .Range("H5" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
509         ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("H5" ).Value
510     End If
511     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
512     ActiveCell.Offset(0, 1).Select
513     End With
514 End Sub
515 Sub DayBtn36()
1
```

```
1
516 With Sheets("CalPopUp" )
517 If .Range("B6" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
518     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("B6" ).Value
519 End If
520     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
521     ActiveCell.Offset(0, 1).Select
522 End With
523 End Sub
524 Sub DayBtn37()
525 With Sheets("CalPopUp" )
526 If .Range("C6" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
527     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("C6" ).Value
528 End If
529     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
530     ActiveCell.Offset(0, 1).Select
531 End With
532 End Sub
533 Sub DayBtn38()
534 With Sheets("CalPopUp" )
535 If .Range("D6" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
536     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("D6" ).Value
537 End If
538     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
539     ActiveCell.Offset(0, 1).Select
540 End With
541 End Sub
542 Sub DayBtn39()
543 With Sheets("CalPopUp" )
544 If .Range("E6" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
545     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("E6" ).Value
546 End If
547     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
548     ActiveCell.Offset(0, 1).Select
549 End With
550 End Sub
551 Sub DayBtn4()
552 With Sheets("CalPopUp" )
553 If .Range("E1" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
554     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("E1" ).Value
555 End If
556     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
557     ActiveCell.Offset(0, 1).Select
558 End With
559 End Sub
560 Sub DayBtn40()
561 With Sheets("CalPopUp" )
562 If .Range("F6" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
563     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("F6" ).Value
564 End If
565     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
566     ActiveCell.Offset(0, 1).Select
567 End With
568 End Sub
569 Sub DayBtn41()
570 With Sheets("CalPopUp" )
571 If .Range("G6" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
572     ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("G6" ).Value
573 End If
574     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
575     ActiveCell.Offset(0, 1).Select
576 End With
```

1

```
1
577 End Sub
578 Sub DayBtn42()
579     With Sheets("CalPopUp" )
580         If .Range("H6" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
581             ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("H6" ).Value
582         End If
583         ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
584         ActiveCell.Offset(0, 1).Select
585     End With
586 End Sub
587 Sub DayBtn5()
588     With Sheets("CalPopUp" )
589         If .Range("F1" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
590             ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("F1" ).Value
591         End If
592         ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
593         ActiveCell.Offset(0, 1).Select
594     End With
595 End Sub
596 Sub DayBtn6()
597     With Sheets("CalPopUp" )
598         If .Range("G1" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
599             ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("G1" ).Value
600         End If
601         ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
602         ActiveCell.Offset(0, 1).Select
603     End With
604 End Sub
605 Sub DayBtn7()
606     With Sheets("CalPopUp" )
607         If .Range("H1" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
608             ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("H1" ).Value
609         End If
610         ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
611         ActiveCell.Offset(0, 1).Select
612     End With
613 End Sub
614 Sub DayBtn8()
615     With Sheets("CalPopUp" )
616         If .Range("B2" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
617             ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("B2" ).Value
618         End If
619         ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
620         ActiveCell.Offset(0, 1).Select
621     End With
622 End Sub
623 Sub DayBtn9()
624     With Sheets("CalPopUp" )
625         If .Range("C2" ).Value <> Empty And .Range("A7" ).Value <> Empty Then
626             ActiveSheet.Range(.Range("A7" ).Value).Value = .Range("C2" ).Value
627         End If
628         ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
629         ActiveCell.Offset(0, 1).Select
630     End With
631 End Sub
632 Sub GroupCal()
633     ActiveSheet.Shapes.Range(Array("NextTri" , "NextRec" )).Group.Select
634     Selection.ShapeRange.Name = "NextMonth"
635     ActiveSheet.Shapes.Range(Array("PrevTri" , "PrevRec" )).Group.Select
636     Selection.ShapeRange.Name = "PrevMonth"
637     ActiveSheet.Shapes.Range(Array("Settings" , "40Day" , "41Day" , "39Day" , "38Day" _
```

1

```

1
638     , "42Day" , "37Day" , "36Day" , "CalBack" , "MonthYear" , "CalBorder" , "1Day" , _
639     "3Day" , "14Day" , "7Day" , "4Day" , "2Day" , "5Day" , "8Day" , "10Day" , "6Day" , _
640     "13Day" , "11Day" , "9Day" , "12Day" , "15Day" , "17Day" , "20Day" , "21Day" ,
    "18Day" _
641     , "16Day" , "19Day" , "22Day" , "24Day" , "26Day" , "27Day" , "25Day" , "23Day" , _
642     "28Day" , "29Day" , "31Day" , "34Day" , "35Day" , "32Day" , "30Day" , "33Day" , "Sa" ,
    _
643     "Fr" , "Th" , "We" , "Tu" , "Mo" , "Su" , "SetBtn" , "CalCol1" , "CalCol2" ,
    "CalCol3" , _
644     "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" , "PrevMonth" ,
    _
645     "NextMonth" )).Visible = msoCTrue
646     ActiveSheet.Shapes.Range(Array("Settings" , "40Day" , "41Day" , "39Day" , "38Day" _
647     , "42Day" , "37Day" , "36Day" , "CalBack" , "MonthYear" , "CalBorder" , "1Day" , _
648     "3Day" , "14Day" , "7Day" , "4Day" , "2Day" , "5Day" , "8Day" , "10Day" , "6Day" , _
649     "13Day" , "11Day" , "9Day" , "12Day" , "15Day" , "17Day" , "20Day" , "21Day" , "18Day" _
650     , "16Day" , "19Day" , "22Day" , "24Day" , "26Day" , "27Day" , "25Day" , "23Day" , _
651     "28Day" , "29Day" , "31Day" , "34Day" , "35Day" , "32Day" , "30Day" , "33Day" , "Sa" , _
652     "Fr" , "Th" , "We" , "Tu" , "Mo" , "Su" , "SetBtn" , "CalCol1" , "CalCol2" , "CalCol3" , _
    "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" , "PrevMonth" , _
653     "NextMonth" )).Select
654     Selection.ShapeRange.Group.Select
655     Selection.ShapeRange.Name = "Calendar"
656     Selection.Name = "Calendar"
657     Selection.Placement = xlMove
658     ActiveSheet.Shapes("Calendar" ).Placement = 2
659 End Sub
660
661
662
663
664
665
666
667
668
669 Sub MacroLinkRemover()
670     'PURPOSE: Remove an external workbook reference from all shapes triggering macros
671     'Source: www.ExcelForFreelancers.com
672     Dim Shp As Shape
673     Dim MacroLink, NewLink As String
674     Dim SplitLink As Variant
675
676     For Each Shp In ActiveSheet.Shapes 'Loop through each shape in worksheet
677
678         'Grab current macro link (if available)
679         On Error GoTo NextShp
680         MacroLink = Shp.OnAction
681
682         'Determine if shape was linking to a macro
683         If MacroLink <> "" And InStr(MacroLink, "!") <> 0 Then
684             'Split Macro Link at the exclamation mark (store in Array)
685             SplitLink = Split(MacroLink, "!")
686
687             'Pull text occurring after exclamation mark
688             NewLink = SplitLink(1)
689
690             'Remove any straggling apostrophes from workbook name
691             If Right(NewLink, 1) = "'" Then
692                 NewLink = Left(NewLink, Len(NewLink) - 1)
693             End If

```

1 2 3

```

1 2 3
694
695     'Apply New Link
696     Shp.OnAction = NewLink
697     End If
698     NextShp:
699     Next Shp
700 End Sub
701 Sub NextMonth()
702     'Next Month button
703     If Sheets("CalPopUp" ).Range("A20" ).Value <> Empty Then
704         DayName = Sheets("CalPopUp" ).Range("A20" ).Value & "Day"
705         With ActiveSheet.Shapes(DayName)
706             .Fill.ForeColor.RGB = RGB(255, 255, 255)
707             .TextFrame2.TextRange.Font.Bold = msoFalse
708         End With
709     End If
710     With Sheets("CalPopUp" )
711         If .Range("A3" ).Value = 12 Then
712             .Range("A3" ).Value = 1
713             .Range("A2" ).Value = .Range("A2" ).Value + 1
714         Else:
715             .Range("A3" ).Value = .Range("A3" ).Value + 1
716         End If
717         If .Range("A6" ).Value > 0 Then
718             ActiveSheet.Shapes.Range(Array("36Day" , "37Day" , "38Day" , "39Day" , "40Day" ,
719                 "41Day" , "42Day" )).Visible = True
720         Else:
721             ActiveSheet.Shapes.Range(Array("36Day" , "37Day" , "38Day" , "39Day" , "40Day" ,
722                 "41Day" , "42Day" )).Visible = False
723         End If
724     End With
725 End Sub
726 Sub PrevMonth()
727     'Previous Month Button
728     If Sheets("CalPopUp" ).Range("A20" ).Value <> Empty Then
729         DayName = Sheets("CalPopUp" ).Range("A20" ).Value & "Day"
730         With ActiveSheet.Shapes(DayName)
731             .Fill.ForeColor.RGB = RGB(255, 255, 255)
732             .TextFrame2.TextRange.Font.Bold = msoFalse
733         End With
734     End If
735     With Sheets("CalPopUp" )
736         If .Range("A3" ).Value = 1 Then
737             .Range("A3" ).Value = 12
738             .Range("A2" ).Value = .Range("A2" ).Value - 1
739         Else:
740             .Range("A3" ).Value = .Range("A3" ).Value - 1
741         End If
742         If .Range("A6" ).Value > 0 Then
743             ActiveSheet.Shapes.Range(Array("36Day" , "37Day" , "38Day" , "39Day" , "40Day" ,
744                 "41Day" , "42Day" )).Visible = True
745         Else:
746             ActiveSheet.Shapes.Range(Array("36Day" , "37Day" , "38Day" , "39Day" , "40Day" ,
747                 "41Day" , "42Day" )).Visible = False
748         End If
749     End With
750 End Sub
751 Sub ShowSettings()
752     'Show or Hide Calendar Settings Panel
753     If ActiveSheet.Shapes.Range(Array("Settings" )).Visible = True Then
754         ActiveSheet.Shapes.Range(Array("Settings" , "CalCol1" , "CalCol2" , "CalCol3" ,

```

1 2

```
1 2  
    "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" )).Visible =  
        False  
751     Else:  
752     ActiveSheet.Shapes.Range(Array("Settings" , "CalCol1" , "CalCol2" , "CalCol3" ,  
        "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" )).Visible = True  
753     End If  
754     End Sub  
755  
756     Sub UnGroupCal()  
757         On Error Resume Next  
758         ActiveSheet.Shapes("Calendar" ).Ungroup  
759         ActiveSheet.Shapes("NextMonth" ).Ungroup  
760         ActiveSheet.Shapes("PrevMonth" ).Ungroup  
761         On Error GoTo 0  
762     End Sub
```

—, 69, 70

A

Activate, 61
 ActiveCell, 61, 63-69
 ActiveSheet, 59-72
 ActiveWorkbook, 61, 62
 ActSht, 61
 Add, 61, 62
 Address, 61, 62
 After, 61
 Array, 59-61, 69-72
 AutoFill, 62, 63

B

Bold, 60, 71

C

CalCol, 61, 62
 CalCol1, 59
 CalCol2, 59
 CalCol3, 59
 CalCol4, 59
 CalCol5, 59
 CalCol6, 59
 CalCol7, 59
 CalCol8, 59
 CalCol9, 60
 CalendarHide, 60
 CalendarShow, 60
 CalFormulaReplacement, 60, 61
 Cells, 61, 62
 CheckForSheet, 61
 Chr, 62, 63
 ColCnt, 61, 62
 ColNum, 61
 Copy, 61
 Count, 61
 CreateCalSht, 61
 CreateWS, 61

D

DayBtn1, 63
 DayBtn10, 63
 DayBtn11, 64
 DayBtn12, 64
 DayBtn13, 64
 DayBtn14, 64
 DayBtn15, 64
 DayBtn16, 64
 DayBtn17, 64
 DayBtn18, 65
 DayBtn19, 65
 DayBtn2, 65
 DayBtn20, 65
 DayBtn21, 65
 DayBtn22, 65
 DayBtn23, 65
 DayBtn24, 66
 DayBtn25, 66
 DayBtn26, 66
 DayBtn27, 66
 DayBtn28, 66
 DayBtn29, 66
 DayBtn3, 66
 DayBtn30, 67
 DayBtn31, 67
 DayBtn32, 67

DayBtn33, 67
 DayBtn34, 67
 DayBtn35, 67
 DayBtn36, 67
 DayBtn37, 68
 DayBtn38, 68
 DayBtn39, 68
 DayBtn4, 68
 DayBtn40, 68
 DayBtn41, 68
 DayBtn42, 69
 DayBtn5, 69
 DayBtn6, 69
 DayBtn7, 69
 DayBtn8, 69
 DayBtn9, 69
 DayCnt, 61, 62
 DayName, 59, 60, 71
 DayNum, 60, 61
 Destination, 62, 63
 DrawingObject, 60-62

E

Empty, 60, 63-69, 71

F

Fill, 59, 60, 71
 Font, 60, 71
 ForeColor, 59, 60, 71
 Formula, 60-62

G

Group, 69, 70
 GroupCal, 60, 63, 69

I

InStr, 60, 70
 IsDate, 60

L

Left, 61, 70
 Len, 70

M

MacroLink, 70
 MacroLinkRemover, 60, 70
 Month, 60
 MsgBox, 61, 63
 msoCTrue, 61, 70
 msoFalse, 60, 63-69, 71
 msoTrue, 60

N

Name, 61, 62, 69, 70
 Names, 62
 NewLink, 70, 71
 NextMonth, 71
 NextShp, 70, 71
 NoCal, 60, 61, 63
 NumberFormat, 63

O

Offset, 61, 63-69
 OnAction, 60, 62, 70, 71

P

Paste, 62
 Placement, 70
 PrevMonth, 71

R

Range, 59-72
 RefersTo, 62
 RGB, 59, 60, 71
 Right, 70
 RowCnt, 61, 62
 RowNum, 61

S

SelCell, 59-61
 Selection, 59, 60, 69, 70
 Shape, 61, 70
 ShapeRange, 59, 60, 69, 70
 Shapes, 59-72
 Sheets, 59-69, 71
 ShowSettings, 71
 Shp, 61, 70, 71
 Split, 70
 SplitLink, 70

T

TextFrame2, 60, 71
 TextRange, 60, 71
 ThisWorkbook, 61
 Top, 61

U

Ungroup, 72
 UnGroupCal, 60, 62, 72

V

Value, 59-69, 71
 Visible, 60-72

W

Worksheet, 61
 ws, 61

X

xlFillDefault, 62, 63
 xlMove, 70
 xlSheetHidden, 62

Y

Year, 60

```

1 Option Explicit
2 Dim ShtNm, ScSum As String
3 Sub AddAttachment()
4     Dim AttFile As FileDialog
5     Set AttFile = Application.FileDialog(msoFileDialogFilePicker)
6     With AttFile
7         .Title = "Select files to attach"
8         If .Show <> -1 Then GoTo NoSelection
9         Sheet1.Range("O" & ActiveCell.Row).Value = Dir(.SelectedItems(1))
10        Sheet1.Range("R" & ActiveCell.Row).Value = .SelectedItems(1)
11    End With
12    NoSelection:
13 End Sub
14
15
16
17 Sub LoadMonth()
18     Dim LastRow, ScRow, ScCol, StCol, EndCol, DbCol, AddRows, FiltRow, LastFiltRow As Long
19     Dim MtgCount As Long
20     Dim FindMonthDates, FoundMonthDates As Range
21     Dim MtgDate As Date
22     StopCalc
23     With Sheet1
24         Set FindMonthDates = .Range("MonthDates" )
25         .Range("D6:J10,D12:J16,D18:J22,D24:J28,D30:J34,D36:J40,BB6:BH40" ).ClearContents
26         Sheet6.Range("AA2:AB2,AA4:AF99999" ).ClearContents 'Clear old Criteria & Results
27         'Sheet6.Range("AC2").Value = ">=" & .Range("B8").Value 'From 1st of Month (these Lines
                are commented out and replaced with formulas)
28         'Sheet6.Range("AD2").Value = "<=" & .Range("B9").Value 'To Last Day of month (these
                Lines are commented out and replaced with formulas)
29         LastRow = Sheet6.Range("C99999" ).End(xlUp).Row
30
31         Sheet6.Range("C3:J" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=Sheet6.
                Range("AC1:AD2" ), CopyToRange:=Sheet6.Range("AA3:AD3" ), unique:=True
32         LastFiltRow = Sheet6.Range("AA99999" ).End(xlUp).Row
33         If LastFiltRow < 4 Then GoTo NoData
34         'Sort By Time, Then By Date
35         Sheet6.Sort.SortFields.Clear
36         Sheet6.Sort.SortFields.Add Key:=Sheet6.Range("AD3" ), SortOn:=xlSortOnValues, Order:=
                xlAscending, DataOption:=xlSortNormal 'Sort By Time
37         With Sheet6.Sort
38             .SetRange Range("AA4:AD" & LastFiltRow)
39             .Apply
40         End With
41         Sheet6.Sort.SortFields.Clear
42         Sheet6.Sort.SortFields.Add Key:=Sheet6.Range("AC3" ), SortOn:=xlSortOnValues, Order:=
                xlAscending, DataOption:=xlSortNormal 'Sort By Time
43         With Sheet6.Sort
44             .SetRange Range("AA4:AD" & LastFiltRow)
45             .Apply
46         End With
47
48         'Load Meetings Into Month
49         For FiltRow = 4 To LastFiltRow
50             MtgDate = Sheet6.Range("AC" & FiltRow).Value
51             Set FoundMonthDates = FindMonthDates.Find(MtgDate, , xlValues, xlWhole)
52             If Not FoundMonthDates Is Nothing Then ' Found Date
53                 ScCol = .Range("AB" & FoundMonthDates.Row).Value
54                 ScRow = .Range("AC" & FoundMonthDates.Row).Value
55                 If MtgCount = 5 Then MtgCount = 4
56                 .Cells(ScRow + MtgCount, ScCol).Value = Application.WorksheetFunction.Text(Sheet6

```

1 2 3 4


```
1 2 3 4
57 .Range("AD" & FiltRow).Value, "h:mm/p" ) & " " & Sheet6.Range("AB" & FiltRow)
    .Value
58 .Cells(ScRow + MtgCount, ScCol + 50).Value = Sheet6.Range("AA" & FiltRow).Value
    If Sheet6.Range("AC" & FiltRow + 1).Value = MtgDate Then MtgCount = MtgCount + 1
        Else: MtgCount = 0
59
60     End If
61 Next FiltRow
62
63     NoData:
64 End With
65 ResetCalc
66 End Sub
67
68 Sub NextMonth_Sched()
69     With Sheet1
70         If .Range("B7" ).Value = 12 Then
71             .Range("B7" ).Value = 1
72             .Range("S4" ).Value = .Range("S4" ).Value + 1
73         Else:
74             .Range("B7" ).Value = .Range("B7" ).Value + 1
75         End If
76         Application.Run "MonthSel" & .Range("B7" ).Value
77     End With
78 End Sub
79
80 Sub PrevMonth_Sched()
81     With Sheet1
82         If .Range("B7" ).Value = 1 Then
83             .Range("B7" ).Value = 12
84             .Range("S4" ).Value = .Range("S4" ).Value - 1
85         Else:
86             .Range("B7" ).Value = .Range("B7" ).Value - 1
87         End If
88         Application.Run "MonthSel" & .Range("B7" ).Value
89     End With
90 End Sub
91
92 Sub PrintMeeting()
93     Sheet1.PrintOut , , , True, True, , , False
94 End Sub
```

A

ActiveCell, [74](#)
Add, [74](#)
AddAttachment, [74](#)
AddRows, [74](#)
AdvancedFilter, [74](#)
Application, [74](#), [75](#)
Apply, [74](#)
AttFile, [74](#)

C

Cells, [74](#), [75](#)
Clear, [74](#)
ClearContents, [74](#)
CopyToRange, [74](#)
CriteriaRange, [74](#)

D

DataOption, [74](#)
DbCol, [74](#)
Dir, [74](#)

E

EndCol, [74](#)
Explicit, [74](#)

F

FileDialog, [74](#)
FiltRow, [74](#), [75](#)
Find, [74](#)
FindMonthDates, [74](#)
FoundMonthDates, [74](#)

K

Key, [74](#)

L

LastFiltRow, [74](#)
LastRow, [74](#)
LoadMonth, [74](#)

M

msoFileDialogFilePicker, [74](#)
MtgCount, [74](#), [75](#)
MtgDate, [74](#), [75](#)

N

NextMonth_Sched, [75](#)
NoData, [74](#), [75](#)
NoSelection, [74](#)

O

Order, [74](#)

P

PrevMonth_Sched, [75](#)
PrintMeeting, [75](#)
PrintOut, [75](#)

R

Range, [74](#), [75](#)
ResetCalc, [75](#)
Row, [74](#)
Run, [75](#)

S

ScCol, [74](#), [75](#)
ScRow, [74](#), [75](#)
ScSum, [74](#)

SelectedItems, [74](#)

SetRange, [74](#)
Sheet1, [74](#), [75](#)
Sheet6, [74](#), [75](#)
Show, [74](#)
ShtNm, [74](#)
Sort, [74](#)
SortFields, [74](#)
SortOn, [74](#)
StCol, [74](#)
StopCalc, [74](#)

T

Text, [74](#)
Title, [74](#)

U

unique, [74](#)

V

Value, [74](#), [75](#)

W

WorksheetFunction, [74](#)

X

xlAscending, [74](#)
xlFilterCopy, [74](#)
xlSortNormal, [74](#)
xlSortOnValues, [74](#)
xlUp, [74](#)
xlValues, [74](#)
xlWhole, [74](#)

```
1 Option Explicit
2
3 Sub ClearAdvancedFilter()
4 'This macro clears named ranges used in Advanced filter. This is important when using
5 multiple advanced filters to clear in between
6 On Error Resume Next 'Skips errors caused when no Criteria or Extract Named Ranges exist
7 With Sheet3 'With Invoice List Sheet
8     .Names("Criteria" ).Delete 'Clears any named ranged for Criteria
9     .Names("Extract" ).Delete 'Clears any named ranged for Extract
10 End With
11 On Error GoTo 0
12 End Sub
13
14 Sub Staff_GetUniquePositions()
15 Dim LastStaffRow, LastFiltRow As Long
16 With Sheet3
17     .Range("Y4:Y9999" ).ClearContents 'Clear Previous Unique Positions
18     LastStaffRow = .Range("C9999" ).End(xlUp).Row
19     ClearAdvancedFilter
20     .Range("C2:G" & LastStaffRow).AdvancedFilter xlFilterCopy, CopyToRange:=.Range("Y3" ),
21     unique:=True
22 End With
23 End Sub
```

A

AdvancedFilter, [77](#)

C

ClearAdvancedFilter, [77](#)

ClearContents, [77](#)

CopyToRange, [77](#)

D

Delete, [77](#)

E

Explicit, [77](#)

L

LastFiltRow, [77](#)

LastStaffRow, [77](#)

N

Names, [77](#)

R

Range, [77](#)

Row, [77](#)

S

Sheet3, [77](#)

Staff_GetUniquePositions, [77](#)

U

unique, [77](#)

X

xlFilterCopy, [77](#)

xlUp, [77](#)

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