

VBA SOURCE CODE BOOK

MOBILE EMPLOYEE TIMESHEET - UPDATED

The screenshot displays the 'MOBILE EMPLOYEE TIMESHEET' application interface. It includes a search bar, a list of employees with their names and contact information, a form for entering time logs, and a summary table of employee time logs. The summary table is as follows:

Date/Time In	Date/Time Out	Hourly Wage	Tot. Reg. Hrs	Tot. OT Hrs	Tot. Regular	Total OT	Total Pay
4/17/22 1:25 PM	4/17/2022 2:25	\$20.00	8.00	1.00	\$160.00	\$80.00	\$190.00
4/18/2022 11:00	4/18/2022 19:00	\$20.00	6.00	0.00	\$120.00	\$0.00	\$120.00
4/23/2022 12:00	4/23/2022 3:00	\$20.00	7.66	0.00	\$153.12	\$0.00	\$153.12

Below the table is a calendar view for April 2022, showing employee time logs for each day. The interface also includes a mobile app preview showing a calendar and employee list.

EXCLUSIVE FOR
SILVER & GOLD
MEMBERS

FEATURE, FIX,
OR FOCUS



Excel For Freelancers

How To Create An Employee Attendance Timesheet WITH Mobile App Sync In Excel



[DOWNLOAD
APPLICATION](#)



[VIEW
TRAINING](#)

by: *Randy Austin*

ABOUT THE AUTHOR

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

With over 300,000 YouTube subscribers, 15,000,000 video views, 200+ comprehensive training videos, and a thriving 50,000 member Facebook community, Excel For Freelancers has positioned itself as the #1 Excel developers resource in the world.

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



[WEBSITE](#)



[YOUTUBE](#)



[FACEBOOK](#)



[TWITTER](#)



[DISCORD](#)



[INSTAGRAM](#)



[TELEGRAM](#)



[RUMBLE](#)



Microsoft®
Most Valuable
Professional



OUR COURSES & PRODUCTS



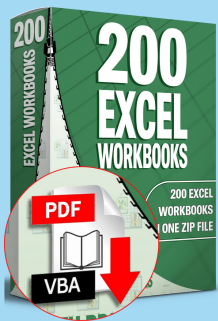
This comprehensive program will take you through a 12-phase process that will turn your enthusiasm for Excel into passive income.

[Click here to learn more](#)



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

[Click here to learn more](#)



Incredible Package of 200 of my BEST Applications now with PDF VBA Codebooks packed into a SINGLE ZIP File which also includes the "200 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](#)

Table of Contents

Projects.....	4
VBAProject.....	4
Documents.....	4
Admin.....	4
(Declarations).....	4
Worksheet_SelectionChange [Sub].....	4
CalPopUp.....	5
(Declarations).....	5
Employees.....	6
(Declarations).....	6
Schedule.....	7
(Declarations).....	7
Sheet1.....	8
(Declarations).....	8
Sheet2.....	9
(Declarations).....	9
ThisWorkbook.....	10
(Declarations).....	10
TimeEntries.....	11
(Declarations).....	11
Worksheet_Change [Sub].....	11
Worksheet_SelectionChange [Sub].....	11
TimesheetDB.....	13
(Declarations).....	13
Modules.....	14
App_Macros.....	14
(Declarations).....	14
Admin_SetBookingColor [Sub].....	14
BrowseForAppDataFolder [Sub].....	14
BrowseForIconFolder [Sub].....	14
BrowseForMobileDataFile [Sub].....	14
UpdateFromAppData [Sub].....	14
Dashboard.....	16
(Declarations).....	16
Dashboard_Refresh [Sub].....	16
PopUp_Calendar.....	17
(Declarations).....	17
CalCol [Sub].....	17
CalendarHide [Sub].....	17
CalendarShow [Sub].....	17
CalFormulaReplacement [Sub].....	18
CheckForSheet [Sub].....	18
CreateCalSht [Sub].....	19
GroupCal [Sub].....	21
MacroLinkRemover [Sub].....	21
NextMonth [Sub].....	22
NextYear [Sub].....	22
PrevMonth [Sub].....	22
PrevYear [Sub].....	23
ReplaceCalendar [Sub].....	23
SelectDay [Sub].....	23
ShowSettings [Sub].....	23
UnGroupCal [Sub].....	24
Schedule_Macros.....	25
(Declarations).....	25
Sched_Entry_Select [Sub].....	25
Sched_NextMonth [Sub].....	25
Sched_PrevMonth [Sub].....	25
Sched_Refresh [Sub].....	25
Sched_ThisMonth [Sub].....	27
Time_Picker_Macros.....	29
(Declarations).....	29
AMPMSel [Sub].....	29
AMSel [Sub].....	29
DoneBtn [Sub].....	29
GrpHrs [Sub].....	29

GrpMin [Sub]	29
GrpSet [Sub]	29
HideHrSel [Sub]	30
HideMinSel [Sub]	30
HoursDisplay [Sub]	30
MacroLinkRemover [Sub]	31
MinDisplay [Sub]	32
PMSel [Sub]	32
ProtTP [Sub]	33
Select1 [Sub]	33
Select10 [Sub]	33
Select11 [Sub]	33
Select12 [Sub]	33
Select2 [Sub]	34
Select3 [Sub]	34
Select4 [Sub]	34
Select5 [Sub]	34
Select6 [Sub]	34
Select7 [Sub]	34
Select8 [Sub]	35
Select9 [Sub]	35
SelectM00 [Sub]	35
SelectM05 [Sub]	35
SelectM10 [Sub]	35
SelectM15 [Sub]	35
SelectM20 [Sub]	35
SelectM25 [Sub]	36
SelectM30 [Sub]	36
SelectM35 [Sub]	36
SelectM40 [Sub]	36
SelectM45 [Sub]	36
SelectM50 [Sub]	36
SelectM55 [Sub]	37
Set10MinInt [Sub]	37
Set15MinInt [Sub]	37
Set5MinInt [Sub]	37
SetHideShow [Sub]	37
SetThm1 [Sub]	37
SetThm2 [Sub]	38
SetThm3 [Sub]	38
SetThm4 [Sub]	38
SetThm5 [Sub]	38
TPShow [Sub]	38
UnGrpHrs [Sub]	39
UnGrpMin [Sub]	39
UnGrpSet [Sub]	39
UnProtTP [Sub]	39
TimeEntry_Macros	40
(Declarations)	40
Empl_SavedMsg [Sub]	40
Employee_LoadList [Sub]	40
Employee_SaveUpdate [Sub]	40
Entry_BrowseEmplPic [Sub]	41
Entry_ClearEmplPic [Sub]	41
Entry_Delete [Sub]	42
Entry_Load [Sub]	42
Entry_New [Sub]	43
Entry_SavedMsg [Sub]	43
Entry_SaveUpdate [Sub]	43
Entry_ShowEmplPic [Sub]	44
TimeEntry_ClearFilter [Sub]	45
TimeEntry_LoadList [Sub]	45
TimeLog_Macros	46
(Declarations)	46
TimeLog_Print [Sub]	46
TimeLog_Refresh [Sub]	46

TimeLog_ShowHide [Sub]46

```
1 Option Explicit
2
3 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
4     If Target.CountLarge > 1 Then Exit Sub
5     If Shapes("ColorPalette").Visible = True Then Shapes("ColorPalette").Visible =
        msoFalse
6
7
8
9     'On Work Day Change
10    If Not Intersect(Target, Range("D19:D25")) Is Nothing Then
11        If Target.Value = Empty Then Target.Value = Chr(252) Else Target.ClearContents
12        Range("D18").Select
13    End If
14
15    'Show color palette
16    If Not Intersect(Target, Range("D11:D14")) Is Nothing And Range("C" & Target.Row).
        Value <> Empty Then
17        With Shapes("ColorPalette")
18            .Left = Range("D" & Target.Row).Left
19            .Top = Range("D" & Target.Row + 1).Top
20            .Visible = msoCTrue
21        End With
22    End If
23 End Sub
```

1 Option Explicit

2

1	Option Explicit
2	

1	Option Explicit
2	

1 Option Explicit

2

1 Option Explicit

2

1 Option Explicit

2

```

1 Option Explicit
2
3 Private Sub Worksheet_Change(ByVal Target As Range)
4     'On Filter change, run Employee List Load
5     If Not Intersect(Target, Range("F3" )) Is Nothing Then Employee_LoadList 'Load
        Employee List
6
7     'On Employee Change, load employee details
8     If Not Intersect(Target, Range("J4" )) Is Nothing And Range("J4" ).Value <> Empty
        And Range("B7" ).Value <> Empty Then
9         Dim EmpRow As Long
10        EmpRow = Range("B7" ).Value
11        Range("J6" ).Value = Target.Value 'Employee Name
12        Range("M6" ).Value = Employees.Range("D" & EmpRow).Value 'Hourly Wage
13        Range("J8" ).Value = Employees.Range("F" & EmpRow).Value 'Email
14        Range("M8" ).Value = Employees.Range("C" & EmpRow).Value 'Employee Type
15        Range("B6" ).Value = Employees.Range("L" & EmpRow).Value 'Employee Picture
16        If Range("B6" ).Value <> "" Then Entry_ShowEmplPic 'Show Employee Picture
17    End If
18
19    'On Time Log Date Change Refresh log
20    If Not Intersect(Target, Range("W3,Y3" )) Is Nothing And Range("W3" ).Value <> Empty
        And Range("Y3" ).Value <> Empty Then TimeLog_Refresh
21 End Sub
22
23 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
24     Dim Cal As Shape, TimePick As Shape
25     'Check To Make Time Picker Exists
26     On Error Resume Next
27     Set TimePick = Shapes("TimePickGrp" )
28     On Error GoTo 0
29     If TimePick Is Nothing Then 'Shape Deleted
30         Application.EnableEvents = False
31         On Error Resume Next
32         Application.Undo
33         On Error GoTo 0
34         Application.EnableEvents = True
35     End If
36     'Check To make sure calendar exists, if not, copy from Developers sheet and paste
        it in
37     On Error Resume Next
38     Set Cal = Shapes("Calendar" )
39     On Error GoTo 0
40     If Cal Is Nothing Then ReplaceCalendar 'Shape Deleted
41     If Shapes("Calendar" ).Visible = True Then CalendarHide 'Run macro to hide calendar
42
43     If Shapes("TimePickGrp" ).Visible = True Then Shapes("TimePickGrp" ).Visible =
        msoFalse 'Hide Time Picker if displayed
44
45     If Target.CountLarge > 1 Then Exit Sub
46
47     'On Employee Selection, load Time Entries
48     If Not Intersect(Target, Range("D5:D999" )) Is Nothing And Range("D" & Target.Row).
        Value <> Empty Then
49         Range("B5" ).Value = Target.Row
50         With Shapes("EmplPntr" )
51             .Left = Range("E" & Target.Row).Left
52             .Top = Range("E" & Target.Row).Top
53             .Visible = msoCTrue
54         End With

```

1 2

```
1 2
54 TimeEntry_LoadList 'Load List of Employee time entries
55 If Range("I:I" ).EntireColumn.Hidden = True Then TimeLog_Refresh 'Refresh Time
    Log If Displayed
56 End If
57
58 'On Timesheet Entry Selection
59 If Not Intersect(Target, Range("F6:G999" )) Is Nothing And Range("F" & Target.Row).
    Value <> Empty Then
60     Range("B9" ).Value = Target.Row
61     Range("B2" ).Value = Range("E" & Target.Row).Value 'Add Entry ID
62     If Range("I:I" ).EntireColumn.Hidden = True Then TimeLog_ShowHide 'Show time
        clock
63     Entry_Load
64 End If
65
66 'On Date Field Selection Show Calendar
67 If Not Intersect(Target, Range("J11,J13,W3,Y3" )) Is Nothing Then 'Change this to
    any cell (or cells) you would like to have the Pop-Up Calendar Appear
68     CheckForSheet
69     CalendarShow
70 Else:
71     CheckForSheet
72     CalendarHide
73 End If
74
75
76 'On Time Field selection
77 If Not Intersect(Target, Range("M11,M13" )) Is Nothing Then
78     'If Sheet is Protected, add Unprotect Code here such as ActiveSheet.Unprotect
        "Password"
79     TPSHow
80 Else:
81     Shapes("TimePickGrp" ).Visible = msoFalse
82 End If
83 End Sub
```

1	Option Explicit
2	


```

1 Option Explicit
2
3 Sub Admin_SetBookingColor()
4     ActiveCell.Interior.Color = Admin.Shapes(Application.Caller).Fill.ForeColor.RGB
5     Admin.Shapes("ColorPalette").Visible = msoFalse
6 End Sub
7
8 Sub BrowseForAppDataFolder()
9     Dim EntryFolder As FileDialog
10    Set EntryFolder = Application.FileDialog(msoFileDialogFolderPicker)
11    With EntryFolder
12        .Title = "Browse For Mobile Data Folder"
13        .AllowMultiSelect = False
14        If .Show <> -1 Then GoTo NoSelection
15        Admin.Range("D5").Value = .SelectedItems(1)
16    End With
17    NoSelection:
18 End Sub
19
20 Sub BrowseForIconFolder()
21     Dim EntryFolder As FileDialog
22     Set EntryFolder = Application.FileDialog(msoFileDialogFolderPicker)
23     With EntryFolder
24         .Title = "Browse For Icon Folder"
25         .AllowMultiSelect = False
26         If .Show <> -1 Then GoTo NoSelection
27         Admin.Range("D7").Value = .SelectedItems(1)
28     End With
29     NoSelection:
30 End Sub
31
32 Sub BrowseForMobileDataFile()
33     Dim DataFile As FileDialog
34     Set DataFile = Application.FileDialog(msoFileDialogFilePicker)
35     With DataFile
36         .Title = "Browse For TimeEntries Data Excel File"
37         .Filters.Add "Select Excel File", "*.xlsx, *.xls", 1
38         .AllowMultiSelect = False
39         If .Show <> -1 Then GoTo NoSelection
40         Admin.Range("D6").Value = .SelectedItems(1)
41     End With
42     NoSelection:
43 End Sub
44
45 Sub UpdateFromAppData()
46     Dim DataFilePath As String, DataFolder As String, FileName As String
47     Dim EntryRow As Long, EntryCol As Long, EntryCount As Long
48     Dim EntryDataArr() As String, EntryData As String, EntryID As String
49     DataFolder = Admin.Range("MobileDataFolder").Value & "\App_Data\"
50     If Dir(DataFolder, vbDirectory) = "" Or DataFolder = "" Then
51         MsgBox "Please select a correct Dropbox Mobile App Data Folder in the Admin
52             screen"
53         Exit Sub
54     End If
55     With TimesheetDB
56         FileName = Dir(DataFolder & "*.txt")
57         Do While Len(FileName) > 0 'Start of Loop
58             DataFilePath = DataFolder & FileName
59             EntryID = Replace(Replace(FileName, "Entry_", ""), ".txt", "") 'Set Entry

```

1 2 3

```
1 2 3
ID
60 On Error Resume Next
61 EntryRow = TimesheetDB.Range("Entry_ID" ).Find(EntryID, , xlValues, xlWhole).
Row 'Get Entry Row
62 On Error GoTo 0
63 If EntryRow = 0 Then
64     EntryRow = .Range("A99999" ).End(xlUp).Row + 1 'First Available Row for
        New Entries
65     .Range("A" & EntryRow).Value = EntryID 'Set Entry ID for New records
66 End If
67 Open DataFilePath For Input As #1
68 Line Input #1, EntryData
69 Close #1
70 EntryDataArr = Split(EntryData, "/" )
71 For EntryCol = 2 To 5
72     .Cells(EntryRow, EntryCol).Value = EntryDataArr(EntryCol - 2) 'Add/Update
        Data
73 Next EntryCol
74 FileName = Dir() 'Clear out Current File name
75 NoEntry:
76 Kill (DataFilePath)
77 EntryCount = EntryCount + 1
78 Loop
79 MsgBox EntryCount & " TimeEntries have been updated or added"
80 End With
81 End Sub
```

```
1 Option Explicit
2
3 Sub Dashboard_Refresh()
4     Dim LastRow As Long, LastResultRow As Long
5     With TimesheetDB
6         LastRow = .Range("A99999").End(xlUp).Row 'Last Row
7         If LastRow < 2 Then Exit Sub
8         .Range("AJ3:AU99999").ClearContents 'Clear Results
9         .Range("A1:H" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
10            "AW1:AX2" ), CopyToRange:=.Range("AJ2:A02" ), Unique:=True
11        LastResultRow = .Range("AJ99999").End(xlUp).Row
12        If LastResultRow < 3 Then Exit Sub
13        .Range("AP3:AU" & LastResultRow).Formula = .Range("AP1:A01").Formula 'Bring
14        over formulas
15    End With
16    ThisWorkbook.RefreshAll
17 End Sub
```

```

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

```

1 2

```

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

```

```

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

```

1 2

```

1 2
168 .Range("A8" ).AutoFill Destination:=.Range("A8:A19" ), Type:=xlFillDefault
169 .Range("A20" ).Value =
    "=IFERROR(INDIRECT(ADDRESS(SUMPRODUCT((B1:H6=A1)*ROW(B1:H6))+6,SUMPRODUCT((B1:H6=A
1)*COLUMN(B1:H6)),1,1))," & Chr(34) & Chr(34) & ")"
170 'Set Defined Names
171 ActiveWorkbook.Names.Add Name:="CalMonths" , RefersTo:="=CalPopUp!$A$8:$A$19"
172 ActiveWorkbook.Names.Add Name:="CalYear" , RefersTo:="=CalPopUp!$A$2"
173
174 'Add in Calendar Formulas
175
176 .Range("B1" ).Value =
    "=IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=1,DATE(CalYear,MATCH($A$4,Cal
177 Months,0),1)," & Chr(34) & Chr(34) & ")"
    .Range("C1" ).Value = "=IF(B1<>" & Chr(34) & Chr(34) &
    ",B1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=2,DATE(CalYear,MATCH($A
178 $4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")"
    .Range("D1" ).Value = "=IF(C1<>" & Chr(34) & Chr(34) &
    ",C1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=3,DATE(CalYear,MATCH($A
179 $4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")"
    .Range("E1" ).Value = "=IF(D1<>" & Chr(34) & Chr(34) &
    ",D1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=4,DATE(CalYear,MATCH($A
180 $4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")"
    .Range("F1" ).Value = "=IF(E1<>" & Chr(34) & Chr(34) &
    ",E1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=5,DATE(CalYear,MATCH($A
181 $4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")"
    .Range("G1" ).Value = "=IF(F1<>" & Chr(34) & Chr(34) &
    ",F1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=6,DATE(CalYear,MATCH($A
182 $4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")"
    .Range("H1" ).Value = "=IF(G1<>" & Chr(34) & Chr(34) &
    ",G1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=7,DATE(CalYear,MATCH($A
183 $4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")"
    .Range("B2" ).Value = "=H1+1"
184 .Range("C2" ).Value = "=B2+1"
185 .Range("C2" ).AutoFill Destination:=.Range("C2:H2" ), Type:=xlFillDefault
186 .Range("B2:H2" ).AutoFill Destination:=.Range("B2:H4" ), Type:=xlFillDefault
187 .Range("B5" ).Value = "=IF(OR(H4=" & Chr(34) & Chr(34) & ",MONTH(H4+1)<>$A$3),"
    & Chr(34) & Chr(34) & ",H4+1)"
188 .Range("C5" ).Value = "=IFERROR(IF(MONTH(B5+1)<>$A$3," & Chr(34) & Chr(34) &
    ",B5+1)," & Chr(34) & Chr(34) & ")"
189 .Range("B6" ).Value = "=IFERROR(IF(OR(H5=" & Chr(34) & Chr(34) &
    ",MONTH(H5+1)<>$A$3)," & Chr(34) & Chr(34) & ",H5+1)," & Chr(34) & Chr(34) &
    ")")
190 .Range("C6" ).Value = "=IFERROR(F(MONTH(I5+1)<>$A$3," & Chr(34) & Chr(34) &
    ",I5+1)," & Chr(34) & Chr(34) & ")"
191 .Range("C5:C6" ).AutoFill Destination:=.Range("C5:H6" ), Type:=xlFillDefault
192
193 'Set format to Single Day
194 .Range("B1:H6" ).NumberFormat = "d"
195
196 'Add in relative Day #'s
197 .Range("B7" ).Value = "1"
198 .Range("C7" ).Value = "2"
199 .Range("B8" ).Value = "8"
200 .Range("C8" ).Value = "9"
201 .Range("B7:C8" ).AutoFill Destination:=.Range("B7:H8" ), Type:=xlFillDefault
202 .Range("B7:H8" ).AutoFill Destination:=.Range("B7:H12" ), Type:=xlFillDefault
203 GroupCal
204 End With
205 Exit Sub
206 NoCal:

```

```

1
207 MsgBox "The Pop-up Calendar does not exist on this worksheet. Please copy the
calendar over from another sheet and paste into this sheet"
208 End Sub
209 Sub GroupCal()
210 ActiveSheet.Shapes.Range(Array("NextTri" , "NextRec" )).Group.Select
211 Selection.ShapeRange.Name = "NextMonth"
212 ActiveSheet.Shapes.Range(Array("PrevTri" , "PrevRec" )).Group.Select
213 Selection.ShapeRange.Name = "PrevMonth"
214 ActiveSheet.Shapes.Range(Array("Settings" , "40Day" , "41Day" , "39Day" , "38Day" _
215 , "42Day" , "37Day" , "36Day" , "CalBack" , "Month" , "Year" , "CalBorder" ,
"1Day" , _
216 "3Day" , "14Day" , "7Day" , "4Day" , "2Day" , "5Day" , "8Day" , "10Day" ,
"6Day" , _
217 "13Day" , "11Day" , "9Day" , "12Day" , "15Day" , "17Day" , "20Day" , "21Day" ,
"18Day" _
218 , "16Day" , "19Day" , "22Day" , "24Day" , "26Day" , "27Day" , "25Day" , "23Day"
, _
219 "28Day" , "29Day" , "31Day" , "34Day" , "35Day" , "32Day" , "30Day" , "33Day" ,
"Sa" , _
220 "Fr" , "Th" , "We" , "Tu" , "Mo" , "Su" , "SetBtn" , "CalCol1" , "CalCol2" ,
"CalCol3" , _
221 "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" ,
"PrevMonth" , _
222 "NextMonth" , "NextYr" , "PrevYr" ).Visible = msoCTrue
223 ActiveSheet.Shapes.Range(Array("Settings" , "40Day" , "41Day" , "39Day" , "38Day" _
224 , "42Day" , "37Day" , "36Day" , "CalBack" , "Month" , "Year" , "CalBorder" , "1Day"
, _
225 "3Day" , "14Day" , "7Day" , "4Day" , "2Day" , "5Day" , "8Day" , "10Day" , "6Day" , _
226 "13Day" , "11Day" , "9Day" , "12Day" , "15Day" , "17Day" , "20Day" , "21Day" ,
"18Day" _
227 , "16Day" , "19Day" , "22Day" , "24Day" , "26Day" , "27Day" , "25Day" , "23Day" , _
228 "28Day" , "29Day" , "31Day" , "34Day" , "35Day" , "32Day" , "30Day" , "33Day" ,
"Sa" , _
229 "Fr" , "Th" , "We" , "Tu" , "Mo" , "Su" , "SetBtn" , "CalCol1" , "CalCol2" ,
"CalCol3" , _
230 "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" , "PrevMonth"
, _
231 "NextMonth" , "NextYr" , "PrevYr" )).Select
232 Selection.ShapeRange.Group.Select
233 Selection.ShapeRange.Name = "Calendar"
234 Selection.Name = "Calendar"
235 Selection.Placement = xlMove
236 ActiveSheet.Shapes("Calendar" ).Placement = 2
237 End Sub
238
239
240
241
242
243
244 Sub MacroLinkRemover()
245 'PURPOSE: Remove an external workbook reference from all shapes triggering macros
246 'Source: www.ExcelForFreelancers.com
247 Dim Shp As Shape
248 Dim MacroLink, NewLink As String
249 Dim SplitLink As Variant
250
1

```



```

1
251 For Each Shp In ActiveSheet.Shapes 'Loop through each shape in worksheet
252
253     'Grab current macro link (if available)
254     On Error GoTo NextShp
255     MacroLink = Shp.OnAction
256
257     'Determine if shape was linking to a macro
258     If MacroLink <> "" And InStr(MacroLink, "!") <> 0 Then
259         'Split Macro Link at the exclamation mark (store in Array)
260         SplitLink = Split(MacroLink, "!")
261
262         'Pull text occurring after exclamation mark
263         NewLink = SplitLink(1)
264
265         'Remove any stragglng apostrophes from workbook name
266         If Right(NewLink, 1) = "'" Then
267             NewLink = Left(NewLink, Len(NewLink) - 1)
268         End If
269
270         'Apply New Link
271         Shp.OnAction = NewLink
272     End If
273     NextShp:
274 Next Shp
275 End Sub
276 Sub NextMonth()
277     'Next Month button
278     If Sheets("CalPopUp").Range("A20").Value <> Empty Then
279         DayName = Sheets("CalPopUp").Range("A20").Value & "Day"
280         With ActiveSheet.Shapes(DayName)
281             .Fill.ForeColor.RGB = RGB(255, 255, 255)
282             .TextFrame2.TextRange.Font.Bold = msoFalse
283         End With
284     End If
285     With Sheets("CalPopUp")
286         If .Range("A3").Value = 12 Then
287             .Range("A3").Value = 1
288             .Range("A2").Value = .Range("A2").Value + 1
289         Else:
290             .Range("A3").Value = .Range("A3").Value + 1
291         End If
292         If .Range("A6").Value > 0 Then
293             ActiveSheet.Shapes.Range(Array("36Day", "37Day", "38Day", "39Day",
294                 "40Day", "41Day", "42Day")).Visible = True
295         Else:
296             ActiveSheet.Shapes.Range(Array("36Day", "37Day", "38Day", "39Day",
297                 "40Day", "41Day", "42Day")).Visible = False
298         End If
299     End With
300 End Sub
301 Sub NextYear()
302     ThisWorkbook.Sheets("CalPopUp").Range("A2").Value = ThisWorkbook.Sheets(
303         "CalPopUp").Range("A2").Value + 1
304 End Sub
305 Sub PrevMonth()
306     'Previous Month Button
307     If Sheets("CalPopUp").Range("A20").Value <> Empty Then
308         DayName = Sheets("CalPopUp").Range("A20").Value & "Day"
309         With ActiveSheet.Shapes(DayName)

```

1 2 3

```

1 2 3
308     .Fill.ForeColor.RGB = RGB(255, 255, 255)
309     .TextFrame2.TextRange.Font.Bold = msoFalse
310     End With
311     End If
312     With Sheets("CalPopUp" )
313     If .Range("A3" ).Value = 1 Then
314         .Range("A3" ).Value = 12
315         .Range("A2" ).Value = .Range("A2" ).Value - 1
316     Else:
317         .Range("A3" ).Value = .Range("A3" ).Value - 1
318     End If
319     If .Range("A6" ).Value > 0 Then
320         ActiveSheet.Shapes.Range(Array("36Day" , "37Day" , "38Day" , "39Day" ,
321             "40Day" , "41Day" , "42Day" )).Visible = True
322     Else:
323         ActiveSheet.Shapes.Range(Array("36Day" , "37Day" , "38Day" , "39Day" ,
324             "40Day" , "41Day" , "42Day" )).Visible = False
325     End If
326     End With
327     End Sub
328
329 Sub PrevYear()
330     ThisWorkbook.Sheets("CalPopUp" ).Range("A2" ).Value = ThisWorkbook.Sheets(
331         "CalPopUp" ).Range("A2" ).Value - 1
332 End Sub
333
334 Sub ReplaceCalendar() 'Shape Deleted
335     ThisWorkbook.Sheets("CalPopUp" ).Shapes("Calendar" ).Copy 'Copy From Developers
336     sheet
337     ActiveCell.Select 'Select the active Cell
338     ActiveSheet.Paste 'Paste in Calendar
339 End Sub
340
341 .....
342 ''''Select Day Of The Month
343 .....
344
345 Sub SelectDay()
346     Dim DayNumb As Long, RowNumb As Long, ColNumb As Long
347     DayNumb = Replace(Application.Caller, "Day" , "" )
348     RowNumb = Application.WorksheetFunction.RoundUp(DayNumb / 7, 0)
349     ColNumb = DayNumb Mod 7 + 1
350     If ColNumb = 1 Then ColNumb = 8
351     'On Error Resume Next
352     If ThisWorkbook.Sheets("CalPopUp" ).Range("A7" ).Value = Empty Then Exit Sub
353     ActiveSheet.Range(ThisWorkbook.Sheets("CalPopUp" ).Range("A7" ).Value).Value =
354     ThisWorkbook.Sheets("CalPopUp" ).Cells(RowNumb, ColNumb).Value
355     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
356     ActiveCell.Offset(0, 1).Select
357 End Sub
358
359 Sub ShowSettings()
360     'Show or Hide Calendar Settings Panel
361     If ActiveSheet.Shapes.Range(Array("Settings" )).Visible = True Then
362         ActiveSheet.Shapes.Range(Array("Settings" , "CalCol1" , "CalCol2" , "CalCol3" ,
363             "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" )).Visible
364             = False
365     Else:
366         ActiveSheet.Shapes.Range(Array("Settings" , "CalCol1" , "CalCol2" , "CalCol3" ,
367             "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" )).Visible
368             = True
369     End If
370 End Sub

```

```
359  
360 Sub UnGroupCal()  
361     On Error Resume Next  
362     ActiveSheet.Shapes("Calendar").Ungroup  
363     ActiveSheet.Shapes("NextMonth").Ungroup  
364     ActiveSheet.Shapes("PrevMonth").Ungroup  
365     On Error GoTo 0  
366 End Sub
```

```
1 Option Explicit
```

```
2
3 Sub Sched_Entry_Select()
```

```
4 Dim EntryID As Long, EntryRow As Long, SelRow As Long
```

```
5 With TimeEntries
```

```
6
7 EntryID = Replace(Application.Caller, Left(Application.Caller, 9), "" )
```

```
8 Application.ScreenUpdating = False
```

```
9 .Range("B2" ).Value = EntryID
```

```
10 .Activate
```

```
11 If .Range("I:I" ).EntireColumn.Hidden = True Then TimeLog_ShowHide 'Show time  
clock Entry if hidden
```

```
12 Entry_Load
```

```
13 On Error Resume Next
```

```
14 On Error Resume Next
```

```
15 SelRow = .Range("D5:D9999" ).Find(.Range("J6" ).Value, , xlValues, xlWhole).Row  
'Set Selected Employee Row
```

```
16 On Error GoTo 0
```

```
17 If SelRow <> 0 Then .Range("D" & SelRow).Select 'Select Employee Row To Trigger  
Time Entry Load
```

```
18 On Error Resume Next
```

```
19 .Range("B9" ).Value = .Range("E6:E9999" ).Find(.Range("B2" ).Value, , xlFormulas,  
xlWhole).Row 'Set Selected Time Entry Row
```

```
20 On Error GoTo 0
```

```
21 On Error GoTo 0
```

```
22 'Application.ScreenUpdating = True
```

```
23 End With
```

```
24 End Sub
```

```
25
26 Sub Sched_NextMonth()
```

```
27 With Schedule
```

```
28 If .Range("B11" ).Value = 9 And .Range("B5" ).Value = 12 Then
```

```
29 MsgBox "You are at the last month and last year of the schedule"
```

```
30 Exit Sub
```

```
31 End If
```

```
32 If .Range("B5" ).Value = 12 Then 'On Current January Month
```

```
33 .Range("B11" ).Value = .Range("B11" ).Value + 1 'Next Year
```

```
34 .Range("B5" ).Value = 1 'Set To January
```

```
35 Else ' Non-December
```

```
36 .Range("B5" ).Value = .Range("B5" ).Value + 1 'Increase Increment Month
```

```
37 End If
```

```
38 Sched_Refresh 'Run Macro To Refresh Schedule
```

```
39 End With
```

```
40 End Sub
```

```
41 Sub Sched_PrevMonth()
```

```
42 With Schedule
```

```
43 If .Range("B11" ).Value = 1 And .Range("B5" ).Value = 1 Then
```

```
44 MsgBox "You are at the first month and first year of the schedule"
```

```
45 Exit Sub
```

```
46 End If
```

```
47 If .Range("B5" ).Value = 1 Then 'On Current January Month
```

```
48 .Range("B11" ).Value = .Range("B11" ).Value - 1 'Prev. Year
```

```
49 .Range("B5" ).Value = 12 'Set To December
```

```
50 Else ' Non-January
```

```
51 .Range("B5" ).Value = .Range("B5" ).Value - 1 'Reduce Increment Month
```

```
52 End If
```

```
53 Sched_Refresh 'Run Macro To Refresh Schedule
```

```
54 End With
```

```
55 End Sub
```

```
56
```

```

57 Sub Sched_Refresh()
58 Dim LastRow As Long, LastResultRow As Long, ResultRow As Long, SchedRow As Long,
SchedCol As Long, SortCol As Long, IconNumb As Long
59 Dim CalRow As Long, CalCol As Long, LastEmplRow As Long
60 Dim EntryColor As String, EntryID As String, EmplName As String, EntryStatus As
String, IconFolder As String
61 Dim IconPath As String, GrpString As String, GrpArr() As String
62 Dim Entryshp As Shape
63 Dim ScDate As Date
64 Dim EntryWidth As Double, EntryLeft As Double
65
66 'Clear all existing Entry shapes
67 For Each Entryshp In Schedule.Shapes
68 On Error Resume Next
69 If InStr(Entryshp.Name, "EntryItem" ) > 0 Then Entryshp.Delete 'Delete Schedule
Shapes
70 If InStr(Entryshp.Name, "Icon" ) > 0 Then Entryshp.Delete 'Delete Icons
71 On Error GoTo 0
72 Next Entryshp
73 Schedule.Range("D4:D999" ).ClearContents 'Clear Existing Employee Names
74 LastEmplRow = Employees.Range("A99999" ).End(xlUp).Row 'Last Employee Row
75 If LastEmplRow < 2 Then Exit Sub 'Exit on no employees
76 Schedule.Range("D4:D" & LastEmplRow + 2).Value = Employees.Range("B2:B" &
LastEmplRow).Value 'Bring over Employee names
77
78 'Get Icons from Admin Screen (if any)
79 IconFolder = Admin.Range("D7" ).Value
80 If IconFolder = Empty Then
81 MsgBox "To dsplay Entry Status Icons, please set an Icon Folder"
82 BrowseForIconFolder
83 If IconFolder = Empty Then GoTo NoIcons
84 End If
85 For IconNumb = 1 To 4
86 Schedule.Range("AJ" & IconNumb + 3).Interior.Color = Admin.Range("D" & IconNumb
+ 10).Interior.Color
87 If Admin.Range("D" & IconNumb + 10).Value <> Empty Then
88 IconPath = IconFolder & "\" & Admin.Range("D" & IconNumb + 10).Value 'Icon
Path
89 If Dir(IconPath, vbDirectory) <> "" Then 'Check For Accurate Path
90 Schedule.Pictures.Insert(IconPath).Name = "Icon" & IconNumb 'Set Icon
91 With Schedule.Shapes("Icon" & IconNumb)
92 .Left = Schedule.Range("AJ" & 3 + IconNumb).Left + 5
93 .Top = Schedule.Range("AJ" & 3 + IconNumb).Top + 1
94 .LockAspectRatio = msoCTrue
95 .Height = 10
96 End With
97 End If
98 End If
99 Next IconNumb
100 NoIcons:
101
102 With TimesheetDB
103 LastRow = .Range("A99999" ).End(xlUp).Row 'Last Row
104 If LastRow < 2 Then Exit Sub
105 .Range("AJ3:A999999" ).ClearContents 'Clear Results
106 .Range("A1:H" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
"AF1:AH2" ), CopyToRange:=.Range("AJ2:A02" ), Unique:=True
107 LastResultRow = .Range("AJ99999" ).End(xlUp).Row
108 If LastResultRow < 3 Then Exit Sub
109 .Range("AP3:AU" & LastResultRow).Formula = .Range("AP1:AU1" ).Formula 'Bring

```

1 2

```

1 2
over formulas
110
111 Application.ScreenUpdating = False
112 For ResultRow = 3 To LastResultRow
113     EntryID = .Range("AJ" & ResultRow).Value 'Entry ID
114     EmplName = .Range("AK" & ResultRow).Value 'Employee Name
115     ScDate = Int(.Range("AM" & ResultRow).Value) 'Entry Date Clock In (*use
Integer to remove time)
116     EntryStatus = .Range("AU" & ResultRow).Value 'Entry Status
117     On Error Resume Next
118     IconNum = Admin.Range("Status" ).Find(EntryStatus, , xlValues, xlWhole).Row -
10 'Get Icon Number from status
119     On Error GoTo 0
120     If IconNum <> 0 Then EntryColor = Admin.Range("D" & IconNum + 10).Interior.
Color ' Set Entry Status Color
121
122     CreateEntry:
123     On Error Resume Next
124     SchedRow = Schedule.Range("D4:D999" ).Find(EmplName, , xlValues, xlWhole).Row
'Get Employee Row
125     On Error GoTo 0
126     On Error Resume Next
127     SchedCol = Day(ScDate) + 4 'Get Schedule Column
128     If SchedRow = 0 Or SchedCol < 5 Then GoTo NextEntry
129
130     Schedule.Shapes("SampleEntryShp" ).Duplicate.Name = "EntryItem" & EntryID
131     With Schedule.Shapes("EntryItem" & EntryID)
132         .Left = Schedule.Cells(SchedRow, SchedCol).Left
133         .Top = Schedule.Cells(SchedRow, SchedCol).Top
134         .Width = Schedule.Cells(SchedRow, SchedCol).Width
135         .Height = Schedule.Cells(SchedRow, SchedCol).Height
136         .OnAction = "Sched_Entry_Select"
137         If EntryColor <> "" Then .Fill.ForeColor.RGB = EntryColor
138     End With
139     GrpString = "EntryItem" & EntryID
140
141     'Add Icons (If applicable)
142     If EntryStatus <> Empty And IconNum <> 0 Then
143         Schedule.Shapes("Icon" & IconNum).Duplicate.Name = "EntryIcon" & EntryID
'Create Icon
144         With Schedule.Shapes("EntryIcon" & EntryID)
145             .Left = Schedule.Cells(SchedRow, SchedCol).Left + (Schedule.Cells(
SchedRow, SchedCol).Width - .Width) / 2
146             .Top = Schedule.Cells(SchedRow, SchedCol).Top + 2
147             .OnAction = "Sched_Entry_Select"
148         End With
149         GrpString = GrpString & "," & "EntryIcon" & EntryID
150     End If
151     SkipIcon:
152
153     'Group Shapes
154     GrpArr = Split(GrpString, ",") 'Create Array of Shapes
155     Schedule.Shapes.Range(GrpArr).Group.Name = "EntryItemGrp" & EntryID
156     GrpString = "" 'Reset Group String
157     NextEntry:
158     Next ResultRow
159 End With
160 Application.ScreenUpdating = True
161 End Sub
162

```

```
163  
164 Sub Sched_ThisMonth()  
165     Schedule.Range("B5").Value = Month(Date)  
166     On Error Resume Next  
167     Schedule.Range("B11").Value = Schedule.Range("Years").Find(Year(Date), ,  
xlFormulas, xlWhole).Row - 17  
168     On Error GoTo 0  
169     Sched_Refresh 'Run Macro To Refresh Schedule  
170 End Sub
```

```

1 Option Explicit
2 Sub AMPMSel()
3     If ActiveSheet.Shapes("AMPM").TextFrame.Characters.Text = "AM" Then PMSel Else:
4     AMSel
5 End Sub
6 Sub AMSel()
7     UnProtTP
8     With ActiveSheet
9         .Shapes("AMPM").TextFrame.Characters.Text = "AM"
10        .Shapes("AMAct").Visible = True
11        .Shapes("AMInact").Visible = False
12        .Shapes("PMAct").Visible = False
13        .Shapes("PMInact").Visible = True
14    End With
15    ProtTP
16 End Sub
17 Sub DoneBtn()
18     With ActiveSheet
19         ActiveCell.Value = .Shapes("HrsDisp").TextFrame.Characters.Text & ":" & .Shapes
20         ("MinDisp").TextFrame.Characters.Text & " " & .Shapes("AMPM").TextFrame.
21         Characters.Text
22         .Shapes("TimePickGrp").Visible = False
23         'SetHideShow
24     End With
25 End Sub
26 Sub GrpHrs()
27     Dim GrpRange, GrpShapes
28     On Error GoTo ExitSub
29     Set GrpShapes = ActiveSheet.Shapes.Range(Array("1Hr", "2Hr", "3Hr", "4Hr",
30     "5Hr", "6Hr", "7Hr", "8Hr", "9Hr", "10Hr", "11Hr", "12Hr" ))
31     Set GrpRange = GrpShapes.Group
32     GrpRange.Name = "HrsGrp"
33     ExitSub:
34 End Sub
35 Sub GrpMin()
36     Dim GrpRange, GrpShapes
37     On Error GoTo ExitSub
38     Set GrpShapes = ActiveSheet.Shapes.Range(Array("00Min", "05Min", "10Min",
39     "15Min", "20Min", "25Min", "30Min", "35Min", "40Min", "45Min", "50Min",
40     "55Min" ))
41     Set GrpRange = GrpShapes.Group
42     GrpRange.Name = "MinGrp"
43     ExitSub:
44 End Sub
45 Sub GrpSet()
46     Dim GrpRange, GrpShapes
47     On Error GoTo ExitSub
48     Set GrpShapes = ActiveSheet.Shapes.Range(Array("5Pie1", "5Pie2" ))
49     Set GrpRange = GrpShapes.Group
50     GrpRange.Name = "5Thm"
51     Set GrpShapes = ActiveSheet.Shapes.Range(Array("4Pie1", "4Pie2" ))
52     Set GrpRange = GrpShapes.Group
53     GrpRange.Name = "4Thm"
54     Set GrpShapes = ActiveSheet.Shapes.Range(Array("3Pie1", "3Pie2" ))
55     Set GrpRange = GrpShapes.Group
56     GrpRange.Name = "3Thm"
57     Set GrpShapes = ActiveSheet.Shapes.Range(Array("2Pie1", "2Pie2" ))
58     Set GrpRange = GrpShapes.Group
59     GrpRange.Name = "2Thm"

```

1


```

1
55 Set GrpShapes = ActiveSheet.Shapes.Range(Array("1Pie1" , "1Pie2" ))
56 Set GrpRange = GrpShapes.Group
57 GrpRange.Name = "1Thm"
58 Set GrpShapes = ActiveSheet.Shapes.Range(Array("5MinPie" , "5MinTxt" ))
59 Set GrpRange = GrpShapes.Group
60 GrpRange.Name = "5MinGrp"
61 Set GrpShapes = ActiveSheet.Shapes.Range(Array("10MinPie" , "10MinTxt" ))
62 Set GrpRange = GrpShapes.Group
63 GrpRange.Name = "10MinGrp"
64 Set GrpShapes = ActiveSheet.Shapes.Range(Array("15MinPie" , "15MinTxt" ))
65 Set GrpRange = GrpShapes.Group
66 GrpRange.Name = "15MinGrp"
67 Set GrpShapes = ActiveSheet.Shapes.Range(Array("SetBack" , "1Thm" , "2Thm" , "3Thm"
, "4Thm" , "5Thm" , "IntTxt" , "ThmTxt" , "5MinGrp" , "10MinGrp" , "15MinGrp" ))
68 Set GrpRange = GrpShapes.Group
69 GrpRange.Name = "SetGrp"
70 ExitSub:
71 End Sub
72 Sub HideHrSel()
73 With ActiveSheet
74 .Shapes("1Sel" ).Visible = False
75 .Shapes("2Sel" ).Visible = False
76 .Shapes("3Sel" ).Visible = False
77 .Shapes("4Sel" ).Visible = False
78 .Shapes("5Sel" ).Visible = False
79 .Shapes("6Sel" ).Visible = False
80 .Shapes("7Sel" ).Visible = False
81 .Shapes("8Sel" ).Visible = False
82 .Shapes("9Sel" ).Visible = False
83 .Shapes("10Sel" ).Visible = False
84 .Shapes("11Sel" ).Visible = False
85 .Shapes("12Sel" ).Visible = False
86 End With
87 End Sub
88 Sub HideMinSel()
89 With ActiveSheet
90 .Shapes("05MSel" ).Visible = False
91 .Shapes("10MSel" ).Visible = False
92 .Shapes("15MSel" ).Visible = False
93 .Shapes("20MSel" ).Visible = False
94 .Shapes("25MSel" ).Visible = False
95 .Shapes("30MSel" ).Visible = False
96 .Shapes("35MSel" ).Visible = False
97 .Shapes("40MSel" ).Visible = False
98 .Shapes("45MSel" ).Visible = False
99 .Shapes("50MSel" ).Visible = False
100 .Shapes("55MSel" ).Visible = False
101 .Shapes("00MSel" ).Visible = False
102 End With
103 End Sub
104 Sub HoursDisplay()
105 Dim Siz, CurLeft, CurTop As Long
106 With ActiveSheet
107 UnProtTP
108 HideMinSel
109 HideHrSel
110 .Shapes("MinDisp" ).TextFrame2.TextRange.Font.Fill.ForeColor.RGB = RGB(255, 255,
255)
111 .Shapes("HrsDisp" ).TextFrame2.TextRange.Font.Fill.ForeColor.RGB = RGB(255, 0, 0)

```

1 2

```

1 2
112 If ActiveCell.Value = Empty Then 'Set Default TIME to 12PM
113     .Shapes("HrsDisp" ).TextFrame.Characters.Text = "12"
114     .Shapes("MinDisp" ).TextFrame.Characters.Text = "00"
115     .Shapes("AMPM" ).TextFrame.Characters.Text = "PM"
116 End If
117     .Shapes("MinGrp" ).Visible = False
118 With .Shapes("HrsGrp" )
119     .Visible = True
120     CurLeft = .Left
121     CurTop = .Top
122     For Siz = 180 To 152 Step -4
123         .Width = Siz
124         .Height = Siz
125         .Left = 152 - Siz + CurLeft
126         .Top = 152 - Siz + CurTop
127         Application.Wait (Now + 0#)
128     Next Siz
129 End With
130 On Error Resume Next
131     .Shapes(.Shapes("HrsDisp" ).TextFrame.Characters.Text & "Sel" ).Visible = True
132 On Error GoTo 0
133 ProtTP
134 End With
135 End Sub
136 Sub MacroLinkRemover()
137     'PURPOSE: Remove an external workbook reference from all shapes triggering macros
138     'Source: www.ExcelForFreelancers.com
139     Dim Shp As Shape
140     Dim MacroLink, NewLink As String
141     Dim SplitLink As Variant
142
143     For Each Shp In ActiveSheet.Shapes 'Loop through each shape in worksheet
144
145         'Grab current macro link (if available)
146         On Error GoTo NextShp
147         MacroLink = Shp.OnAction
148
149         'Determine if shape was linking to a macro
150         If MacroLink <> "" And InStr(MacroLink, "!" ) <> 0 Then
151             'Split Macro Link at the exclamation mark (store in Array)
152             SplitLink = Split(MacroLink, "!" )
153
154             'Pull text occurring after exclamation mark
155             NewLink = SplitLink(1)
156
157             'Remove any stragglng apostrophes from workbook name
158             If Right(NewLink, 1) = "'" Then
159                 NewLink = Left(NewLink, Len(NewLink) - 1)
160             End If
161
162             'Apply New Link
163             Shp.OnAction = NewLink
164         End If
165     NextShp:
166 Next Shp
167 End Sub
168
169
170
171

```

```

172
173 Sub MinDisplay()
174     Dim Siz, CurLeft, CurTop As Long
175     Dim Inter As String
176     Dim Intg5, Intg10, Intg15 As Integer
177     With ActiveSheet
178         UnProtTP
179         Intg5 = .Shapes("5MinTxt").Fill.Visible
180         Intg10 = .Shapes("10MinTxt").Fill.Visible
181         Intg15 = .Shapes("15MinTxt").Fill.Visible
182         If Intg5 = -1 Then Inter = "5M"
183         If Intg10 = -1 Then Inter = "10M"
184         If Intg15 = -1 Then Inter = "15M"
185
186         HideHrSel
187         .Shapes("HrsDisp").TextFrame2.TextRange.Font.Fill.ForeColor.RGB = RGB(255, 255, 255)
188         .Shapes("MInDisp").TextFrame2.TextRange.Font.Fill.ForeColor.RGB = RGB(255, 0, 0)
189
190         .Shapes("HrsGrp").Visible = False
191         .Shapes("MinGrp").Visible = True
192         If Inter = "10M" Or Inter = "15M" Then .Shapes("05Min").Visible = False
193         If Inter = "15M" Then .Shapes("10Min").Visible = False
194         If Inter = "10M" Then .Shapes("15Min").Visible = False
195         If Inter = "15M" Then .Shapes("20Min").Visible = False
196         If Inter = "10M" Or Inter = "15M" Then .Shapes("25Min").Visible = False
197         If Inter = "10M" Or Inter = "15M" Then .Shapes("35Min").Visible = False
198         If Inter = "15M" Then .Shapes("40Min").Visible = False
199         If Inter = "10M" Then .Shapes("45Min").Visible = False
200         If Inter = "15M" Then .Shapes("50Min").Visible = False
201         If Inter = "10M" Or Inter = "15M" Then .Shapes("55Min").Visible = False
202         With .Shapes("MinGrp")
203             .Visible = True
204             CurLeft = .Left
205             CurTop = .Top
206             For Siz = 126 To 152 Step 2
207                 .Width = Siz
208                 .Height = Siz
209                 .Left = 152 - Siz + CurLeft
210                 .Top = 152 - Siz + CurTop
211                 Application.Wait (Now + 0#)
212             Next Siz
213         End With
214         On Error Resume Next
215         .Shapes(.Shapes("MinDisp").TextFrame.Characters.Text & "MSel").Visible = True
216         On Error GoTo 0
217         ProtTP
218     End With
219 End Sub
220
221 Sub PMSel()
222     UnProtTP
223     With ActiveSheet
224         .Shapes("AMPM").TextFrame.Characters.Text = "PM"
225         .Shapes("AMAct").Visible = False
226         .Shapes("AMInact").Visible = True
227         .Shapes("PMAct").Visible = True
228         .Shapes("PMInact").Visible = False
229     End With
230     ProtTP
231 End Sub

```

```

230 Sub ProtTP()
231     Dim GrpRange, GrpShapes
232     On Error GoTo ExitSub
233     Set GrpShapes = ActiveSheet.Shapes.Range(Array("MainBack" , "BackCircle" , "HrsGrp"
234         , "8Sel" , _
235             "9Sel" , "10Sel" , "11Sel" , "12Sel" , "1Sel" , "2Sel" , "3Sel" , "4Sel" ,
236             "5Sel" , _
237             "HrsDisp" , "Colon" , "MinDisp" , "AMPM" , "AMAct" , "PMInact" , "DivLine" ,
238             "DoneBtn" , _
239             "SettingsBtn" , "6Sel" , "7Sel" , "MinGrp" , "PMAct" , "AMInact" ,
240             "00MSel" , _
241             "05MSel" , "10MSel" , "15MSel" , "20MSel" , "25MSel" , "30MSel" , "35MSel" ,
242             "40MSel" , _
243             "45MSel" , "50MSel" , "55MSel" , "SetGrp" ))
244     Set GrpRange = GrpShapes.Group
245     GrpRange.Name = "TimePickGrp"
246     ActiveSheet.Shapes("TimePickGrp" ).Placement = 2
247     ExitSub:
248     'Add in any Protect sheet code here such as ActiveSheet.Protect "Password"
249     'Application.ScreenUpdating = True
250 End Sub

251 Sub Select1()
252     UnProtTP
253     HideHrSel
254     With ActiveSheet
255         .Shapes("1Sel" ).Visible = True
256         .Shapes("HrsDisp" ).TextFrame.Characters.Text = "1"
257         MinDisplay
258     End With
259     ProtTP
260 End Sub

261 Sub Select10()
262     UnProtTP
263     HideHrSel
264     With ActiveSheet
265         .Shapes("10Sel" ).Visible = True
266         .Shapes("HrsDisp" ).TextFrame.Characters.Text = "10"
267         MinDisplay
268     End With
269     ProtTP
270 End Sub

271 Sub Select11()
272     UnProtTP
273     HideHrSel
274     With ActiveSheet
275         .Shapes("11Sel" ).Visible = True
276         .Shapes("HrsDisp" ).TextFrame.Characters.Text = "11"
277         MinDisplay
278     End With
279     ProtTP
280 End Sub

281 Sub Select12()
282     UnProtTP
283     HideHrSel
284     With ActiveSheet
285         .Shapes("12Sel" ).Visible = True
286         .Shapes("HrsDisp" ).TextFrame.Characters.Text = "12"
287         MinDisplay
288     End With

```

1

```
1
285     ProtTP
286 End Sub
287 Sub Select2()
288     UnProtTP
289     HideHrSel
290     With ActiveSheet
291         .Shapes("2Sel" ).Visible = True
292         .Shapes("HrsDisp" ).TextFrame.Characters.Text = "2"
293         If .Shapes("HrsGrp" ).Visible = True Then MinDisplay
294     End With
295     ProtTP
296 End Sub
297 Sub Select3()
298     UnProtTP
299     HideHrSel
300     With ActiveSheet
301         .Shapes("3Sel" ).Visible = True
302         .Shapes("HrsDisp" ).TextFrame.Characters.Text = "3"
303         MinDisplay
304     End With
305     ProtTP
306 End Sub
307 Sub Select4()
308     UnProtTP
309     HideHrSel
310     With ActiveSheet
311         .Shapes("4Sel" ).Visible = True
312         .Shapes("HrsDisp" ).TextFrame.Characters.Text = "4"
313         MinDisplay
314     End With
315     ProtTP
316 End Sub
317 Sub Select5()
318     UnProtTP
319     HideHrSel
320     With ActiveSheet
321         .Shapes("5Sel" ).Visible = True
322         .Shapes("HrsDisp" ).TextFrame.Characters.Text = "5"
323         MinDisplay
324     End With
325     ProtTP
326 End Sub
327 Sub Select6()
328     UnProtTP
329     HideHrSel
330     With ActiveSheet
331         .Shapes("6Sel" ).Visible = True
332         .Shapes("HrsDisp" ).TextFrame.Characters.Text = "6"
333         MinDisplay
334     End With
335     ProtTP
336 End Sub
337 Sub Select7()
338     UnProtTP
339     HideHrSel
340     With ActiveSheet
341         .Shapes("7Sel" ).Visible = True
342         .Shapes("HrsDisp" ).TextFrame.Characters.Text = "7"
343         MinDisplay
344     End With
1
```

```
1
345     ProtTP
346 End Sub
347 Sub Select8()
348     UnProtTP
349     HideHrSel
350     With ActiveSheet
351         .Shapes("8Sel" ).Visible = True
352         .Shapes("HrsDisp" ).TextFrame.Characters.Text = "8"
353         MinDisplay
354     End With
355     ProtTP
356 End Sub
357 Sub Select9()
358     UnProtTP
359     HideHrSel
360     With ActiveSheet
361         .Shapes("9Sel" ).Visible = True
362         .Shapes("HrsDisp" ).TextFrame.Characters.Text = "9"
363         MinDisplay
364     End With
365     ProtTP
366 End Sub
367 Sub SelectM00()
368     UnProtTP
369     HideMinSel
370     With ActiveSheet
371         .Shapes("00MSel" ).Visible = True
372         .Shapes("MinDisp" ).TextFrame.Characters.Text = "00"
373     End With
374     ProtTP
375 End Sub
376 Sub SelectM05()
377     UnProtTP
378     HideMinSel
379     With ActiveSheet
380         .Shapes("05MSel" ).Visible = True
381         .Shapes("MinDisp" ).TextFrame.Characters.Text = "05"
382     End With
383     ProtTP
384 End Sub
385 Sub SelectM10()
386     UnProtTP
387     HideMinSel
388     With ActiveSheet
389         .Shapes("10MSel" ).Visible = True
390         .Shapes("MinDisp" ).TextFrame.Characters.Text = "10"
391     End With
392     ProtTP
393 End Sub
394 Sub SelectM15()
395     UnProtTP
396     HideMinSel
397     With ActiveSheet
398         .Shapes("15MSel" ).Visible = True
399         .Shapes("MinDisp" ).TextFrame.Characters.Text = "15"
400     End With
401     ProtTP
402 End Sub
403 Sub SelectM20()
404     UnProtTP
```

1

```
1
405 HideMinSel
406 With ActiveSheet
407     .Shapes("20MSel").Visible = True
408     .Shapes("MinDisp").TextFrame.Characters.Text = "20"
409 End With
410 ProtTP
411 End Sub
412 Sub SelectM25()
413     UnProtTP
414     HideMinSel
415     With ActiveSheet
416         .Shapes("25MSel").Visible = True
417         .Shapes("MinDisp").TextFrame.Characters.Text = "25"
418     End With
419     ProtTP
420 End Sub
421 Sub SelectM30()
422     UnProtTP
423     HideMinSel
424     With ActiveSheet
425         .Shapes("30MSel").Visible = True
426         .Shapes("MinDisp").TextFrame.Characters.Text = "30"
427     End With
428     ProtTP
429 End Sub
430 Sub SelectM35()
431     UnProtTP
432     HideMinSel
433     With ActiveSheet
434         .Shapes("35MSel").Visible = True
435         .Shapes("MinDisp").TextFrame.Characters.Text = "35"
436     End With
437     ProtTP
438 End Sub
439 Sub SelectM40()
440     UnProtTP
441     HideMinSel
442     With ActiveSheet
443         .Shapes("40MSel").Visible = True
444         .Shapes("MinDisp").TextFrame.Characters.Text = "40"
445     End With
446     ProtTP
447 End Sub
448 Sub SelectM45()
449     UnProtTP
450     HideMinSel
451     With ActiveSheet
452         .Shapes("45MSel").Visible = True
453         .Shapes("MinDisp").TextFrame.Characters.Text = "45"
454     End With
455     ProtTP
456 End Sub
457 Sub SelectM50()
458     UnProtTP
459     HideMinSel
460     With ActiveSheet
461         .Shapes("50MSel").Visible = True
462         .Shapes("MinDisp").TextFrame.Characters.Text = "50"
463     End With
464     ProtTP
```

1

```
1
465 End Sub
466 Sub SelectM55()
467     UnProtTP
468     HideMinSel
469     With ActiveSheet
470         .Shapes("55MSel").Visible = True
471         .Shapes("MinDisp").TextFrame.Characters.Text = "55"
472     End With
473     ProtTP
474 End Sub
475
476 Sub Set10MinInt()
477     With ActiveSheet
478         .Shapes("10MinTxt").Fill.ForeColor.RGB = RGB(140, 58, 58)
479         .Shapes("10MinTxt").Fill.Visible = msoCTrue
480         .Shapes("5MinTxt").Fill.Visible = msoFalse
481         .Shapes("15MinTxt").Fill.Visible = msoFalse
482     End With
483     MinDisplay
484     SelectM00
485 End Sub
486
487 Sub Set15MinInt()
488     With ActiveSheet
489         .Shapes("15MinTxt").Fill.ForeColor.RGB = RGB(140, 58, 58)
490         .Shapes("15MinTxt").Fill.Visible = msoCTrue
491         .Shapes("10MinTxt").Fill.Visible = msoFalse
492         .Shapes("5MinTxt").Fill.Visible = msoFalse
493     End With
494     MinDisplay
495     SelectM00
496 End Sub
497
498 Sub Set5MinInt()
499     With ActiveSheet
500         .Shapes("5MinTxt").Fill.ForeColor.RGB = RGB(140, 58, 58)
501         .Shapes("5MinTxt").Fill.Visible = msoCTrue
502         .Shapes("10MinTxt").Fill.Visible = msoFalse
503         .Shapes("15MinTxt").Fill.Visible = msoFalse
504     End With
505     MinDisplay
506     SelectM00
507 End Sub
508
509 Sub SetHideShow()
510     With ActiveSheet
511         UnProtTP
512         If .Shapes("SetGrp").Visible = False Then
513             .Shapes("SetGrp").Visible = msoCTrue
514         Else
515             .Shapes("SetGrp").Visible = False
516         End If
517     End With
518     ProtTP
519 End Sub
520
521 Sub SetThm1()
522     ActiveSheet.Shapes.Range(Array("SetBack", "MainBack")).Fill.ForeColor.RGB = RGB(64
523     , 64, 64)
524     ActiveSheet.Shapes("BackCircle").Fill.ForeColor.RGB = RGB(54, 54, 54)
1
```



```

1
524 End Sub
525 Sub SetThm2()
526     ActiveSheet.Shapes.Range(Array("SetBack" , "MainBack" )).Fill.ForeColor.RGB = RGB(55
        , 96, 146)
527     ActiveSheet.Shapes("BackCircle" ).Fill.ForeColor.RGB = RGB(37, 64, 97)
528 End Sub
529 Sub SetThm3()
530     ActiveSheet.Shapes.Range(Array("SetBack" , "MainBack" )).Fill.ForeColor.RGB = RGB(
        119, 147, 60)
531     ActiveSheet.Shapes("BackCircle" ).Fill.ForeColor.RGB = RGB(79, 98, 40)
532 End Sub
533 Sub SetThm4()
534     ActiveSheet.Shapes.Range(Array("SetBack" , "MainBack" )).Fill.ForeColor.RGB = RGB(96
        , 74, 123)
535     ActiveSheet.Shapes("BackCircle" ).Fill.ForeColor.RGB = RGB(64, 49, 82)
536 End Sub
537 Sub SetThm5()
538     ActiveSheet.Shapes.Range(Array("SetBack" , "MainBack" )).Fill.ForeColor.RGB = RGB(49
        , 133, 156)
539     ActiveSheet.Shapes("BackCircle" ).Fill.ForeColor.RGB = RGB(33, 89, 104)
540 End Sub
541 Sub TPShow()
542     Dim SelCell As Range
543     Dim Hr As Long
544     'On Error GoTo NoTP
545     'If Sheet is Protected, add Unprotect Code here such as ActiveSheet.Unprotect
        "Password"
546     With ActiveSheet.Shapes("TimePickGrp" )
547         Set SelCell = Selection
548         .Visible = msoCTrue
549         On Error Resume Next
550         .Left = SelCell.Left
551         .Top = SelCell.Offset(1, 0).Top
552         ' On Error GoTo 0
553     End With
554     UnProtTP
555     HideHrSel
556     HideMinSel
557     'Check for Links from another workbook
558     UnGrpSet
559     If InStr(ActiveSheet.Shapes("5Pie1" ).OnAction, "!" ) <> 0 Then 'Run Workbook Link
        Remover
560         UnGrpHrs
561         UnGrpMin
562         MacroLinkRemover
563         GrpHrs
564         GrpMin
565         GrpSet
566     Else:
567         GrpSet
568     End If
569     If ActiveCell.Value <> Empty Then Hr = Format(ActiveCell.Value, "hh" )
570     If Hr >= 13 Then Hr = Hr - 12
571     If Hr = 0 Then Hr = 12
572
573     If ActiveCell.Value <> Empty Then ActiveSheet.Shapes("HrsDisp" ).TextFrame.
        Characters.Text = Hr
574     If ActiveCell.Value <> Empty Then ActiveSheet.Shapes("MInDisp" ).TextFrame.
        Characters.Text = Right(Format(ActiveCell.Value, "h:mm" ), 2)
575     If ActiveCell.Value <> Empty Then ActiveSheet.Shapes("AMPM" ).TextFrame.Characters.

```

1

```
1
Text = Format(ActiveCell.Value, "AM/PM" )
576 ActiveSheet.Shapes("SetGrp" ).Visible = msoFalse
577 HoursDisplay
578 ProtTP
579 'If Sheet is Protected, Add Protection Code here such as ActiveSheet.Protect
    "Password"
580 Exit Sub
581 NoTP:
582 MsgBox "The Time Picker has been removed from this sheet. Please copy the entire
    Time Picker from another sheet and paste it into this sheet"
583 'If Sheet is Protected, Add Protection Code here such as ActiveSheet.Protect
    "Password"
584 End Sub
585
586 Sub UnGrpHrs()
587     On Error Resume Next
588     ActiveSheet.Shapes("HrsGrp" ).Ungroup
589     On Error GoTo 0
590 End Sub
591
592 Sub UnGrpMin()
593     On Error Resume Next
594     ActiveSheet.Shapes("MinGrp" ).Ungroup
595     On Error GoTo 0
596 End Sub
597
598 Sub UnGrpSet()
599     On Error Resume Next
600     ActiveSheet.Shapes("SetGrp" ).Ungroup
601     ActiveSheet.Shapes("5Thm" ).Ungroup
602     ActiveSheet.Shapes("4Thm" ).Ungroup
603     ActiveSheet.Shapes("3Thm" ).Ungroup
604     ActiveSheet.Shapes("2Thm" ).Ungroup
605     ActiveSheet.Shapes("1Thm" ).Ungroup
606     ActiveSheet.Shapes("5MinGrp" ).Ungroup
607     ActiveSheet.Shapes("10MinGrp" ).Ungroup
608     ActiveSheet.Shapes("15MinGrp" ).Ungroup
609     On Error GoTo 0
610 End Sub
611
612 Sub UnProtTP()
613     On Error Resume Next
614     ActiveSheet.Shapes("TimePickGrp" ).Ungroup
615     On Error GoTo 0
616     'Add in any Unprotect sheet code here such as ActiveSheet.Unprotect "Password"
        'Application.ScreenUpdating = False
End Sub
```

```

1 Option Explicit
2 Dim EntryRow As Long, LastRow As Long, SelRow As Long, LastResultRow As Long, ResultRow
  As Long, DataWkBkRow As Long
3 Dim PicFolder As String, FilePath As String, DataWkBkPath As String, DataFolder As
  String
4 Dim EmplPic As FileDialog
5 Dim DataWkBk As Workbook
6 Sub Empl_SavedMsg()
7     With TimeEntries.Shapes("EmplSavedMsg" )
8         Dim i As Long, Delay As Double, StartTime As Double
9         .Visible = msoCTrue
10        For i = 1 To 150
11            .Fill.Transparency = i / 150
12            Delay = 0.009
13            StartTime = Timer
14            Do
15                DoEvents
16            Loop While Timer - StartTime < Delay
17        Next i
18        .Visible = msoFalse
19    End With
20 End Sub
21 Sub Employee_LoadList()
22     TimeEntries.Range("B5,D5:D9999" ).ClearContents 'Clear Existing Employees
23     With TimesheetDB
24         LastRow = .Range("A99999" ).End(xlUp).Row
25         If LastRow < 2 Then Exit Sub
26         .Range("A1:C" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
27             "K1:K2" ), CopyToRange:=.Range("L1" ), Unique:=True
28         LastResultRow = .Range("L99999" ).End(xlUp).Row
29         If LastResultRow < 2 Then Exit Sub
30         TimeEntries.Range("D5:D" & LastResultRow + 3).Value = .Range("L2:L" &
31             LastResultRow).Value 'Bring Employee Name
32         On Error Resume Next
33         TimeEntries.Range("B5" ).Value = TimeEntries.Range("D5:D999" ).Find(TimeEntries.
34             Range("J6" ).Value, , xlValues, xlWhole).Row
35         On Error GoTo 0
36         If TimeEntries.Range("B5" ).Value <> Empty Then TimeEntries.Range("D" &
37             TimeEntries.Range("B5" ).Value).Select 'Select Employee to reload list
38     End With
39 End Sub
40 Sub Employee_SaveUpdate()
41     Dim EmplRow As Long
42     'Get Mobile Data File Workbook Details
43     DataWkBkPath = [DataFile] 'Get Mobile Data File
44     If DataWkBkPath <> "" And Dir(DataWkBkPath, vbDirectory) <> "" Then
45         Set DataWkBk = Workbooks.Open(DataWkBkPath) 'Set Workbook
46     End If
47     With TimeEntries
48         If .Range("B10" ).Value = Empty Then 'New Employee
49             EmplRow = Employees.Range("A99999" ).End(xlUp).Row + 1 'First Avail Row
50             Employees.Range("A" & EmplRow).Value = .Range("B8" ).Value 'Next Empl ID
51             'Add New Entry To Data Workbook
52             If Not DataWkBk Is Nothing Then
53                 DataWkBk.Sheets("Employees" ).Range("A" & EmplRow).Value = Employees.Range
54                 ("A" & EmplRow).Value 'Set Empl ID
55             End If
56         Else 'Existing Employee

```

1 2 3

```

1 2 3
54     EmplRow = .Range("B10" ).Value 'Existing Employee Row
55     End If
56     Employees.Range("B" & EmplRow).Value = .Range("J6" ).Value 'Employee Name
57     Employees.Range("C" & EmplRow).Value = .Range("M8" ).Value 'Employee Type
58     Employees.Range("D" & EmplRow).Value = .Range("M6" ).Value 'Hourly Wage
59     Employees.Range("F" & EmplRow).Value = .Range("J8" ).Value 'Email
60     Employees.Range("L" & EmplRow).Value = .Range("B6" ).Value 'Picture
61
62     'Update Mobile Data Workbook
63     If Not DataWkBk Is Nothing Then
64         DataWkBk.Sheets("Employees" ).Range("B" & EmplRow).Value = .Range("J6" ).
        Value 'Employee Name
65         DataWkBk.Sheets("Employees" ).Range("C" & EmplRow).Value = .Range("M8" ).
        Value 'Employee Type
66         DataWkBk.Sheets("Employees" ).Range("D" & EmplRow).Value = .Range("M6" ).
        Value 'Hourly Wage
67         DataWkBk.Sheets("Employees" ).Range("F" & EmplRow).Value = .Range("J8" ).
        Value 'Email
68         DataWkBk.Sheets("Employees" ).Range("L" & EmplRow).Value = .Range("B6" ).
        Value 'Picture
69         DataWkBk.Close True 'Save & Close Data Workbook
70     End If
71 End With
72     Empl_SavedMsg
73 End Sub
74
75 Sub Entry_BrowseEmplPic()
76     On Error Resume Next
77     TimeEntries.Shapes("EmplPic" ).Delete 'Remove picture if existing
78     On Error GoTo 0
79
80     'Check for Mobile Data Folder
81     If Admin.Range("MobileDataFolder" ).Value = Empty Then
82         MsgBox "Please set a folder for the mobile data "
83         BrowseForAppDataFolder 'Run Macro to browse for Data folder
84         If Admin.Range("MobileDataFolder" ).Value = Empty Then Exit Sub 'Exit on Empty
85     End If
86     PicFolder = Admin.Range("MobileDataFolder" ).Value & "\\Employees_Images\"
        'Employee Picture Folder
87     Set EmplPic = Application.FileDialog(msoFileDialogFilePicker)
88
89     With EmplPic
90         .Title = "Please select an Employee Picture"
91         .Filters.Add "Picture Files" , "*.jpg,*.png,*.gif, *.jpeg *.GIF" , 1
92         .AllowMultiSelect = False
93         If .Show <> -1 Then GoTo NoSelection
94         If .SelectedItems(1) = PicFolder & Dir(.SelectedItems(1)) Then GoTo SkipCopy
95         If Dir(PicFolder & Dir(.SelectedItems(1)), vbDirectory) <> "" Then Kill (
        PicFolder & Dir(.SelectedItems(1))) 'Delete if existing
96         FileCopy .SelectedItems(1), PicFolder & Dir(.SelectedItems(1)) 'Copy Picture
        into Entry folder
97         SkipCopy:
98         TimeEntries.Range("B6" ).Value = "Employees_Images/" & Dir(.SelectedItems(1))
        'Place file name
99     End With
100     Entry_ShowEmplPic 'Run Macro to Display Entry Picture
101     NoSelection:
102 End Sub
103
104 Sub Entry_ClearEmplPic()

```

```

1
105 On Error Resume Next
106 TimeEntries.Shapes("EmplPic").Delete 'Remove picture if existing
107 On Error GoTo 0
108 TimeEntries.Range("B6").ClearContents
109 End Sub
110
111 Sub Entry_Delete()
112 If MsgBox("Are you sure you want to delete this Entry?" , vbYesNo, "Delete Entry" )
113 = vbNo Then Exit Sub
114 With TimeEntries
115     If .Range("B3").Value = Empty Then GoTo NotSaved
116     EntryRow = .Range("B3").Value 'Entry Row
117     If EntryRow <> 2 Then TimesheetDB.Range(EntryRow & ":" & EntryRow).EntireRow.
118     Delete
119     'Get Mobile Data File Workbook Details
120     DataWkBkPath = [DataFile] 'Get Mobile Data File
121     If DataWkBkPath <> "" And Dir(DataWkBkPath, vbDirectory) <> "" Then
122         Set DataWkBk = Workbooks.Open(DataWkBkPath) 'Set Workbook
123     End If
124     If Not DataWkBk Is Nothing Then 'Clear From Data workbook
125         DataWkBk.Sheets("Timesheet").Range(EntryRow & ":" & EntryRow).EntireRow.
126         Delete
127     End If
128     DataWkBk.Close True 'Save & Close Data Workbook
129     NotSaved:
130     Entry_New 'Clear Out All Form Fields
131     TimeEntry_LoadList 'Reload Entry list
132 End With
133 End Sub
134
135 Sub Entry_Load()
136 With TimeEntries
137     .Range("B6,J4,J6,M6,J8,M8,J11,M11,J13,M13,J15").ClearContents
138     On Error Resume Next
139     .Shapes("EmplPic").Delete 'Remove picture if existing
140     On Error GoTo 0
141     If .Range("B3").Value = Empty Then
142         MsgBox "Please make sure to select a correct Entry"
143         Exit Sub
144     End If
145     Application.ScreenUpdating = False
146     EntryRow = .Range("B3").Value 'Entry Row
147     'Load Employee details
148     .Range("J4").Value = TimesheetDB.Range("B" & EntryRow).Value 'Employee Name
149     (triggers Employee Load)
150     .Range("J15").Value = TimesheetDB.Range("C" & EntryRow).Value 'Project
151     If TimesheetDB.Range("D" & EntryRow).Value <> "" Then 'If Date/Time In is not
152     Empty
153         .Range("J11").Value = Int(TimesheetDB.Range("D" & EntryRow).Value) 'Add
154         Date In (whole #)
155         .Range("M11").Value = TimesheetDB.Range("D" & EntryRow).Value - .Range(
156         "J11").Value 'Time IN
157     End If
158     If TimesheetDB.Range("E" & EntryRow).Value <> "" Then 'If Date/Time Out is not
159     empty
160         .Range("J13").Value = Int(TimesheetDB.Range("E" & EntryRow).Value) 'Add
161         Date in (whole Number)
162         .Range("M13").Value = TimesheetDB.Range("E" & EntryRow).Value - .Range(

```

```

1 2 3
156     "J13" ).Value 'Get Time In
157     End If
158     End With
159     Application.ScreenUpdating = True
160 End Sub
161
162 Sub Entry_New()
163     With TimeEntries
164         .Range("B2,B6,B9,J4,J6,M6,J8,M8,J11,M11,J13,M13,J15" ).ClearContents
165         On Error Resume Next
166         .Shapes("EmplPic" ).Delete
167         On Error GoTo 0
168         .Range("J4" ).Select
169     End With
170 End Sub
171
172 Sub Entry_SavedMsg()
173     With TimeEntries.Shapes("EntrySavedMsg" )
174         Dim i As Long, Delay As Double, StartTime As Double
175         .Visible = msoCTrue
176         For i = 1 To 150
177             .Fill.Transparency = i / 150
178             Delay = 0.009
179             StartTime = Timer
180             Do
181                 DoEvents
182             Loop While Timer - StartTime < Delay
183         Next i
184         .Visible = msoFalse
185     End With
186 End Sub
187
188 Sub Entry_SaveUpdate()
189     With TimeEntries
190         If .Range("J6" ).Value = Empty Then
191             MsgBox "Please make sure to add an Employee before saving this Entry"
192             Exit Sub
193         End If
194
195         'Get Mobile Data File Workbook Details
196         DataWkBkPath = [DataFile] 'Get Mobile Data File
197         If DataWkBkPath <> "" And Dir(DataWkBkPath, vbDirectory) <> "" Then
198             Set DataWkBk = Workbooks.Open(DataWkBkPath) 'Set Workbook
199         End If
200
201         If .Range("B3" ).Value = Empty Then 'New Entry
202             EntryRow = TimesheetDB.Range("A99999" ).End(xlUp).Row + 1 'New Database Row
203             .Range("B2" ).Value = .Range("B4" ).Value 'Set Next Entry ID
204             TimesheetDB.Range("A" & EntryRow).Value = .Range("B2" ).Value 'Set Entry ID
205             Application.ScreenUpdating = False
206             'Add New Entry To Data Workbook
207             If Not DataWkBk Is Nothing Then
208                 DataWkBk.Sheets("Timesheet" ).Range("A" & EntryRow).Value = .Range("B2" ).
209                 Value ' Set Entry ID
210             End If
211         Else 'Existing Entry
212             EntryRow = .Range("B3" ).Value 'Existing Entry Row
213         End If
214         'Add/Update Data on Local Workbook

```

1 2

```

1 2
214 TimesheetDB.Range("B" & EntryRow).Value = .Range("J6").Value 'Employee Name
215 TimesheetDB.Range("C" & EntryRow).Value = .Range("J15").Value 'Project
216 If .Range("B12").Value <> 0 Then TimesheetDB.Range("D" & EntryRow).Value = .
Range("B12").Value 'Date/Time In
217 If .Range("B13").Value <> 0 Then TimesheetDB.Range("E" & EntryRow).Value = .
Range("B13").Value 'Date/Time Out
218 TimesheetDB.Range("F" & EntryRow).Value = .Range("P13").Value + .Range("P15").
Value 'Total Hours
219 TimesheetDB.Range("G" & EntryRow).Value = .Range("M6").Value 'Reg. Hourly
Wage
220 TimesheetDB.Range("H" & EntryRow).Value = .Range("M15").Value 'Total Wages
221
222 'Add/Update Data on App Database
223 If Not DataWkBk Is Nothing Then
224 DataWkBk.Sheets("Timesheet").Range("B" & EntryRow).Value = .Range("J6").
Value 'Employee Name
225 DataWkBk.Sheets("Timesheet").Range("C" & EntryRow).Value = .Range("J15").
Value 'Project
226 If .Range("B12").Value <> 0 Then DataWkBk.Sheets("Timesheet").Range("D" &
EntryRow).Value = .Range("B12").Value 'Date/Time In
227 If .Range("B13").Value <> 0 Then DataWkBk.Sheets("Timesheet").Range("E" &
EntryRow).Value = .Range("B13").Value 'Date/Time Out
228 DataWkBk.Sheets("Timesheet").Range("F" & EntryRow).Value = .Range("P13").
Value + .Range("P15").Value 'Total Hours
229 DataWkBk.Sheets("Timesheet").Range("G" & EntryRow).Value = .Range("M6").
Value 'Reg. Hourly Wage
230 DataWkBk.Sheets("Timesheet").Range("H" & EntryRow).Value = .Range("M15").
Value 'Total Wages
231 End If
232
233 DataWkBk.Close True 'Save & Close Data Workbook
234
235 On Error Resume Next
236 SelRow = .Range("D5:D9999").Find(.Range("J6").Value, , xlValues, xlWhole).Row
'Set Selected Employee Row
237 On Error GoTo 0
238 If SelRow <> 0 Then .Range("D" & SelRow).Select 'Select Employee Row To Trigger
Time Entry Load
239 'TimeEntry_LoadList 'Reload List
240 On Error Resume Next
241 .Range("B9").Value = .Range("E6:E9999").Find(.Range("B2").Value, , xlFormulas,
xlWhole).Row 'Set Selected Time Entry Row
242 On Error GoTo 0
243 Application.ScreenUpdating = True
244 Entry_SavedMsg 'Run Fade Out Message
245 End With
246 End Sub
247 Sub Entry_ShowEmplPic()
248 With TimeEntries
249 On Error Resume Next
250 .Shapes("EmplPic").Delete 'Remove picture if existing
251 On Error GoTo 0
252
253 'Check for Mobile Data Folder
254 If Admin.Range("MobileDataFolder").Value = Empty Then
255 MsgBox "Please set a folder for the mobile data "
256 BrowseForAppDataFolder 'Run Macro to browse for Data folder
257 If Admin.Range("MobileDataFolder").Value = Empty Then Exit Sub 'Exit on
Empty
258 End If

```

```

1 2
259 PicFolder = Admin.Range("MobileDataFolder" ).Value & "\" 'Data Folder
260 FilePath = PicFolder & Replace(.Range("B6" ).Value, "/" , "\" ) 'Picture File
    Path (replace forward slash with back slash)
261
262     If Dir(FilePath, vbDirectory) <> "" Then 'Check For Accurate Path
263         .Pictures.Insert(FilePath).Name = "EmplPic"
264         With .Shapes("EmplPic" )
265             .LockAspectRatio = msoCTrue
266             If .Width > .Height Then .Width = 90 Else .Height = 80 'Set Width / Height
267
268             .Left = TimeEntries.Range("O5" ).Left + (TimeEntries.Range("O:P" ).Width -
                .Width) / 2 'Center Pic Horizontally
269             .Top = TimeEntries.Range("O5" ).Top + (TimeEntries.Range("5:10" ).Height -
                .Height) / 2 'Center Pic Vertically
270             .Visible = msoCTrue
271         End With
272     End If
273 End With
274 End Sub
275 Sub TimeEntry_ClearFilter()
276     TimeEntries.Range("F3" ).Value = "Enter Search"
277 End Sub
278 Sub TimeEntry_LoadList()
279     TimeEntries.Range("B9,E6:G9999" ).ClearContents 'Clear Existing Time Entries
280     With TimesheetDB
281         LastRow = .Range("A99999" ).End(xlUp).Row
282         If LastRow < 2 Then Exit Sub
283         .Range("A1:H" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
                "N1:N2" ), CopyToRange:=.Range("P1:R1" ), Unique:=True
284         LastResultRow = .Range("P99999" ).End(xlUp).Row
285         If LastResultRow < 2 Then Exit Sub
286         With .Sort 'Sort Based on Enter ID (Most recent First)
287             .SortFields.Clear
288             .SortFields.Add Key:=TimesheetDB.Range("P2" ), SortOn:=xlSortOnValues, Order:=
                xlDescending, DataOption:=xlSortNormal 'Sort By Due Date
289             .SetRange TimesheetDB.Range("P2:R" & LastResultRow)
290             .Apply
291         End With
292         TimeEntries.Range("E6:G" & LastResultRow + 4).Value = .Range("P2:R" &
                LastResultRow).Value 'Bring in Log ID & Time Entries
293     End With
294 End Sub

```



```

1 Option Explicit
2 Dim LastRow As Long, LastResultRow As Long
3
4 Sub TimeLog_Print()
5     With TimeEntries
6         LastRow = .Range("S99999" ).End(xlUp).Row
7         .PageSetup.PrintArea = "$S$4:$Z$" & LastRow
8         .PrintOut , , , True, , , False
9     End With
10 End Sub
11
12 Sub TimeLog_Refresh()
13     TimeEntries.Range("S6:Z9999" ).ClearContents
14     With TimesheetDB
15         .Range("W3:AD9999" ).ClearContents 'Clear Previous Results
16         LastRow = .Range("A99999" ).End(xlUp).Row
17         If LastRow < 2 Then Exit Sub
18         .Range("A1:H" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
19             "S1:U2" ), CopyToRange:=.Range("W2:Y2" ), Unique:=True
20         LastResultRow = .Range("W99999" ).End(xlUp).Row
21         If LastResultRow < 3 Then Exit Sub
22         .Range("Z3:AD" & LastResultRow).Formula = .Range("Z1:AD1" ).Formula
23         TimeEntries.Range("S6:Z" & LastResultRow + 3).Value = .Range("W3:AD" &
24             LastResultRow).Value
25     End With
26 End Sub
27 Sub TimeLog_ShowHide()
28     With TimeEntries
29         If .Range("I:I" ).EntireColumn.Hidden = False Then 'Show Time Log & Hide Entries
30             .Shapes("TimeEntryGrp" ).Visible = msoFalse
31             On Error Resume Next
32             .Shapes("EmplPic" ).Visible = msoFalse 'Hide Empl. pic (if any)
33             On Error GoTo 0
34             .Shapes("TimeLogGrp" ).Visible = msoCTrue 'Show Time Log Groip
35             .Range("I:R" ).EntireColumn.Hidden = True 'Hide Time Entry Columns
36             .Range("S:Z" ).EntireColumn.Hidden = False 'Show Time Log Columns
37             TimeLog_Refresh
38         Else 'Hide Time Log & Show Entries
39             .Shapes("TimeEntryGrp" ).Visible = msoCTrue
40             On Error Resume Next
41             .Shapes("EmplPic" ).Visible = msoCTrue 'Show Empl. pic (if any)
42             On Error GoTo 0
43             .Shapes("TimeLogGrp" ).Visible = msoFalse 'Hide Time Log Groip
44             .Range("I:R" ).EntireColumn.Hidden = False 'Show Time Entry Columns
45             .Range("S:Z" ).EntireColumn.Hidden = True 'HideTime Log Columns
46         End If
47     End With
48 End Sub

```

—, 21, 33

A

Activate, 19, 25
 ActiveCell, 14, 17, 18, 23, 29, 31, 38, 39
 ActiveSheet, 17-19, 21-24, 29-39
 ActiveWorkbook, 18-20
 ActSht, 19
 Add, 14, 19, 20, 41, 45
 Address, 18, 19
 Admin, 14, 26, 27, 41, 44, 45
 Admin_SetBookingColor, 14
 AdvancedFilter, 16, 26, 40, 45, 46
 After, 19
 AllowMultiSelect, 14, 41
 AMPMSel, 29
 AMSel, 29
 Application, 11, 14, 17, 23, 25, 27, 31, 32, 41-44
 Apply, 45
 Array, 17, 18, 21-23, 29, 30, 33, 37, 38
 AutoFill, 20

B

Bold, 17, 18, 22, 23
 BrowseForAppDataFolder, 14, 41, 44
 BrowseForIconFolder, 14, 26
 BrowseForMobileDataFile, 14

C

Cal, 11
 CalCol, 17, 19, 26
 CalendarHide, 11, 12, 17
 CalendarShow, 12, 17
 CalFormulaReplacement, 18
 Caller, 14, 17, 23, 25
 CalRow, 26
 Cells, 15, 18, 19, 23, 27
 Characters, 29, 31-38
 CheckForSheet, 12, 18
 Chr, 4, 19, 20
 Clear, 45
 ClearContents, 4, 16, 26, 40, 42, 43, 45, 46
 Close, 15, 41, 42, 44
 ColCnt, 19
 ColNum, 18
 ColNumb, 23
 Color, 14, 26, 27
 Copy, 19, 23
 CopyToRange, 16, 26, 40, 45, 46
 Count, 19
 CountLarge, 4, 11
 CreateCalSht, 19
 CreateEntry, 27
 CreateWS, 18, 19
 CriteriaRange, 16, 26, 40, 45, 46
 CurLeft, 30-32
 CurTop, 30-32

D

Dashboard_Refresh, 16
 DataFile, 14, 40, 42, 43
 DataFilePath, 14, 15
 DataFolder, 14, 40
 DataOption, 45
 DataWkBk, 40-44
 DataWkBkPath, 40, 42, 43
 DataWkBkRow, 40

Day, 18, 27
 DayCnt, 19
 DayName, 17, 18, 22
 DayNum, 17, 18
 DayNumb, 23
 Delay, 40, 43
 Delete, 26, 41-44
 Destination, 20
 Dir, 14, 15, 26, 40-43, 45
 DoEvents, 40, 43
 DoneBtn, 29
 DrawingObject, 18, 19
 Duplicate, 27

E

Empl_SavedMsg, 40, 41
 EmplName, 26, 27
 Employee_LoadList, 11, 40
 Employee_SaveUpdate, 40
 Employees, 11, 26, 40, 41
 EmplPic, 40, 41
 EmplRow, 40, 41
 EmpRow, 11
 Empty, 4, 11, 12, 17, 22, 23, 26, 27, 31, 38, 40-44
 EnableEvents, 11
 EntireColumn, 12, 25, 46
 EntireRow, 42
 Entry_BrowseEmplPic, 41
 Entry_ClearEmplPic, 41
 Entry_Delete, 42
 Entry_Load, 12, 25, 42
 Entry_New, 42, 43
 Entry_SavedMsg, 43, 44
 Entry_SaveUpdate, 43
 Entry_ShowEmplPic, 11, 41, 44
 EntryCol, 14, 15
 EntryColor, 26, 27
 EntryCount, 14, 15
 EntryData, 14, 15
 EntryDataArr, 14, 15
 EntryFolder, 14
 EntryID, 14, 15, 25-27
 EntryLeft, 26
 EntryRow, 14, 15, 25, 40, 42-44
 Entryshp, 26
 EntryStatus, 26, 27
 EntryWidth, 26
 ExitSub, 29, 30, 33
 Explicit, 4-11, 13, 14, 16, 25, 29, 40, 46

F

FileCopy, 41
 FileDialog, 14, 40, 41
 FileName, 14, 15
 FilePath, 40, 45
 Fill, 14, 17, 18, 22, 23, 27, 30, 32, 37, 38, 40, 43
 Filters, 14, 41
 Find, 15, 25, 27, 28, 40, 44
 Font, 17, 18, 22, 23, 30, 32
 ForeColor, 14, 17, 18, 22, 23, 27, 30, 32, 37, 38
 Format, 38, 39
 Formula, 16, 18, 19, 26, 46

G

Group, 21, 27, 29, 30, 33
 GroupCal, 20, 21
 GrpArr, 26, 27

GrpHrs, 29, 38
 GrpMin, 29, 38
 GrpRange, 29, 30, 33
 GrpSet, 29, 38
 GrpShapes, 29, 30, 33
 GrpString, 26, 27

H

Height, 26, 27, 31, 32, 45
 Hidden, 12, 25, 46
 HideHrSel, 30, 32-35, 38
 HideMinSel, 30, 35-38
 HoursDisplay, 30, 39
 Hr, 38

I

i, 40, 43
 IconFolder, 26
 IconNumb, 26, 27
 IconPath, 26
 Input, 15
 Insert, 26, 45
 InStr, 18, 22, 26, 31, 38
 Int, 27, 42
 Inter, 32
 Interior, 14, 26, 27
 Intersect, 4, 11, 12
 Intg10, 32
 Intg15, 32
 Intg5, 32
 IsDate, 17

K

Key, 45
 Kill, 15, 41

L

LastEmplRow, 26
 LastResultRow, 16, 26, 27, 40, 45, 46
 LastRow, 16, 26, 40, 45, 46
 Left, 4, 11, 18, 22, 25-27, 31, 32, 38, 45
 Len, 14, 22, 31
 Line, 15
 LockAspectRatio, 26, 45

M

MacroLink, 21, 22, 31
 MacroLinkRemover, 18, 21, 31, 38
 MinDisplay, 32-35, 37
 Month, 17, 19, 28
 MsgBox, 14, 15, 18, 21, 25, 26, 39, 41-44
 msoCTrue, 4, 11, 18, 21, 26, 37, 38, 40, 43, 45, 46
 msoFalse, 4, 11, 12, 14, 17, 22, 23, 37, 39, 40, 43, 46
 msoFileDialogFilePicker, 14, 41
 msoFileDialogFolderPicker, 14
 msoTrue, 18

N

Name, 19-21, 26, 27, 29, 30, 33, 45
 Names, 20
 NewLink, 21, 22, 31
 NextEntry, 27
 NextMonth, 22
 NextShp, 22, 31
 NextYear, 22
 NoCal, 17, 18, 20
 NoEntry, 15
 Nolcons, 26

NoSelection, 14, 41
 NoTP, 39
 NotSaved, 42
 Now, 31, 32
 NumberFormat, 20

O

Offset, 18, 23, 38
 OnAction, 19, 22, 27, 31, 38
 Open, 15, 40, 42, 43
 Order, 45

P

PageSetup, 46
 Paste, 19, 23
 PicFolder, 40, 41, 45
 Pictures, 26, 45
 Placement, 18, 21, 33
 PMSel, 29, 32
 PrevMonth, 22
 PrevYear, 23
 PrintArea, 46
 PrintOut, 46
 ProtTP, 29, 31-37, 39

R

Range, 4, 11, 12, 14-23, 25-30, 33, 37, 38, 40-46
 RefersTo, 20
 RefreshAll, 16
 Replace, 14, 23, 25, 45
 ReplaceCalendar, 11, 23
 ResultRow, 26, 27, 40
 RGB, 14, 17, 18, 22, 23, 27, 30, 32, 37, 38
 Right, 22, 31, 38
 RoundUp, 23
 Row, 4, 11, 12, 15, 16, 25-28, 40, 43-46
 RowCnt, 19
 RowNum, 18
 RowNumb, 23

S

ScDate, 26, 27
 Sched_Entry_Select, 25
 Sched_NextMonth, 25
 Sched_PrevMonth, 25
 Sched_Refresh, 25, 26, 28
 Sched_ThisMonth, 28
 SchedCol, 26, 27
 SchedRow, 26, 27
 Schedule, 25-28
 ScreenUpdating, 25, 27, 42-44
 SelCell, 17, 18, 38
 Select1, 33
 Select10, 33
 Select11, 33
 Select12, 33
 Select2, 34
 Select3, 34
 Select4, 34
 Select5, 34
 Select6, 34
 Select7, 34
 Select8, 35
 Select9, 35
 SelectDay, 23
 SelectedItems, 14, 41
 Selection, 17, 21, 38
 SelectM00, 35, 37
 SelectM05, 35

SelectM10, 35
 SelectM15, 35
 SelectM20, 35
 SelectM25, 36
 SelectM30, 36
 SelectM35, 36
 SelectM40, 36
 SelectM45, 36
 SelectM50, 36
 SelectM55, 37
 SelRow, 25, 40, 44
 Set10MinInt, 37
 Set15MinInt, 37
 Set5MinInt, 37
 SetHideShow, 37
 SetRange, 45
 SetThm1, 37
 SetThm2, 38
 SetThm3, 38
 SetThm4, 38
 SetThm5, 38
 Shape, 11, 18, 21, 26, 31
 ShapeRange, 17, 21
 Shapes, 4, 11, 12, 14, 17-19, 21-24, 26, 27, 29-46
 Sheets, 17-19, 22, 23, 40-44
 Show, 14, 41
 ShowSettings, 23
 Shp, 18, 21, 22, 31
 Siz, 30-32
 SkipCopy, 41
 SkipIcon, 27
 Sort, 45
 SortCol, 26
 SortFields, 45
 SortOn, 45
 Split, 15, 22, 27, 31
 SplitLink, 21, 22, 31
 StartTime, 40, 43

T

Target, 4, 11, 12
 Text, 29, 31-39
 TextFrame, 29, 31-38
 TextFrame2, 17, 18, 22, 23, 30, 32
 TextRange, 17, 18, 22, 23, 30, 32
 ThisWorkbook, 16, 19, 22, 23
 TimeEntries, 25, 40-46
 TimeEntry_ClearFilter, 45
 TimeEntry_LoadList, 12, 42, 45
 TimeLog_Print, 46
 TimeLog_Refresh, 11, 12, 46
 TimeLog_ShowHide, 12, 25, 46
 TimePick, 11
 Timer, 40, 43
 TimesheetDB, 14-16, 26, 40, 42-46
 Title, 14, 41
 Top, 4, 11, 18, 26, 27, 31, 32, 38, 45
 TPSHOW, 12, 38
 Transparency, 40, 43

U

Undo, 11
 Ungroup, 24, 39
 UnGroupCal, 19, 24
 UnGrpHrs, 38, 39
 UnGrpMin, 38, 39
 UnGrpSet, 38, 39
 Unique, 16, 26, 40, 45, 46
 UnProtTP, 29, 30, 32-39

UpdateFromAppData, 14

V

Value, 4, 11, 12, 14, 15, 17-20, 22, 23, 25-29, 31, 38-46
 vbDirectory, 14, 26, 40-43, 45
 vbNo, 42
 vbYesNo, 42
 Visible, 4, 11, 12, 14, 17-19, 21-23, 29-40, 43, 45, 46

W

Wait, 31, 32
 Width, 27, 31, 32, 45
 Workbook, 40
 Workbooks, 40, 42, 43
 Worksheet, 18, 19
 Worksheet_Change, 11
 Worksheet_SelectionChange, 4, 11
 WorksheetFunction, 23
 ws, 18, 19

X

xlDescending, 45
 xlFillDefault, 20
 xlFilterCopy, 16, 26, 40, 45, 46
 xlFormulas, 25, 28, 44
 xlMove, 18, 21
 xlSheetHidden, 19
 xlSortNormal, 45
 xlSortOnValues, 45
 xlUp, 15, 16, 26, 40, 43, 45, 46
 xlValues, 15, 25, 27, 40, 44
 xlWhole, 15, 25, 27, 28, 40, 44

Y

Year, 17, 19, 28

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