

VBA SOURCE CODE BOOK

HOME RENOVATION PLANNER

PROJECTS & RENOVATIONS

PROJECT NAME	STATUS
Backyard Renovation	Completed
Bathroom Remodel	In Progress
New Guest House	Scheduled
Room Addition	In Progress
Summer Landscape	Delayed
Pool	Completed

JOB DETAILS

Job Name: Design & Scope Job Type: Design

Vendor: C & C Builders Vendor Type: Framing

Contact: Lisa Johnson Cell Phone: (212) 555-4567

Email: lctbuilders@example.com

Budget: \$2,545.00 Actual Cost: \$1,500.00

Start Date: 6/20/2023 Sched. Days: 15

Status: Completed Job Comp. %: 60%

Notes: Test Notes

PROJECT COMPLETION %

Incomplete Jobs: 33% Total Jobs Complete

BUDGETED VS. ACTUAL

Budgeted: \$41,025 Over Budget: \$575 20% Under Budget 20% Above Budget

FREE EXCEL TEMPLATE!



Excel For Freelancers

How To Create A Home Renovation Planner Application In Excel



**DOWNLOAD
APPLICATION**



**VIEW
TRAINING**

by: Randy Austin

ABOUT THE AUTHOR

A 4-time Microsoft MVP & lifetime Excel enthusiast, Randy Austin founded Excel For Freelancers in 2017. Excel For Freelancers quickly became the most prominent resource Excel for developers to learn how to turn their passion for Excel into profits by building & selling their own excel-based applications for passive & recurring income.

With over 471,000 YouTube subscribers, 35,448,742 video views, 430+ comprehensive training videos, and a thriving 65,000 member Facebook community, Excel For Freelancers has positioned itself as the #1 Excel developers resource in the world.

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



[WEBSITE](#)



[YOUTUBE](#)



[FACEBOOK](#)



[TWITTER](#)



[DISCORD](#)



[INSTAGRAM](#)



[TELEGRAM](#)



[RUMBLE](#)



Microsoft[®]
Most Valuable
Professional



OUR COURSES & PRODUCTS



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

[Click here to learn more](#)



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

[Click here to learn more](#)

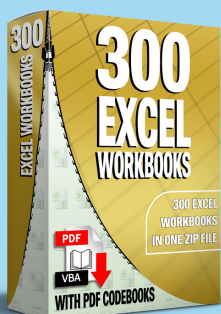
-



This incredible 13-hour freelancer masterclass will teach anyone how to be a successful freelancer with my proven 9-Phase 'Financial Freedom Roadmap' and includes 30+ downloads and exercises.

[Click here to learn more](#)

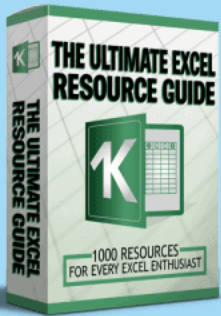
-



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

[Click here to learn more](#)

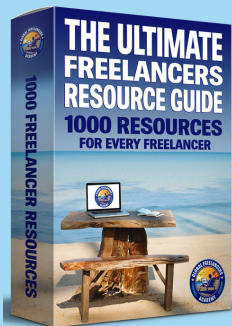
OUR COURSES & PRODUCTS



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

[Click here to learn more](#)

-



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

[Click here to learn more](#)

-



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

[Click here to learn more](#)

Projects.....	2
VBAProject.....	2
Documents.....	2
Admin.....	2
(Declarations).....	2
Worksheet_SelectionChange [Sub].....	2
JobAttach.....	3
(Declarations).....	3
Projects.....	4
(Declarations).....	4
ProjJobs.....	5
(Declarations).....	5
RenPlan.....	6
(Declarations).....	6
Worksheet_Change [Sub].....	6
Worksheet_SelectionChange [Sub].....	6
ThisWorkbook.....	7
(Declarations).....	7
Vendors.....	8
(Declarations).....	8
Modules.....	9
Admin_Macros.....	9
(Declarations).....	9
Admin_SetStatusTypeColor [Sub].....	9
BrowseForAttachFolder [Sub].....	9
Attach_Macros.....	10
(Declarations).....	10
Attach_Add [Sub].....	10
Attach_Delete [Sub].....	10
Attach_DisplayThumbnail [Sub].....	11
Attach_Open [Sub].....	11
Attach_Refresh [Sub].....	12
ProjJob_Macros.....	13
(Declarations).....	13
Job_Delete [Sub].....	13
Job_Load [Sub].....	13
Job_New [Sub].....	13
Job_SaveUpdate [Sub].....	13
Project_AddNew [Sub].....	14
Project_ListJobs [Sub].....	14
Projects_LoadProjects [Sub].....	15
Sched_Macros.....	16
(Declarations).....	16
Sched_JobSelect [Sub].....	16
Sched_NextMonth [Sub].....	16
Sched_PrevMonth [Sub].....	16
Sched_ThisMonth [Sub].....	16
Schedule_Refresh [Sub].....	16

```
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     'Show color palette
7     If Not Intersect(Target, Range("F8:F18")) Is Nothing And Range("E" & Target.Row).
        Value <> Empty Then
8         With Shapes("ColorPalette")
9             .Left = Range("F" & Target.Row).Left
10            .Top = Range("F" & Target.Row + 1).Top
11            .Visible = msoCTrue
12        End With
13    End If
14 End Sub
```

1
2

Option Explicit

1
2

Option Explicit

1 Option Explicit

2

```
1 Option Explicit
2
3 Private Sub Worksheet_Change(ByVal Target As Range)
4     'On Vendor Change, Load Vendor Details
5     If Not Intersect(Target, Range("J7" )) Is Nothing And Range("J7" ).Value <> Empty
6         And Range("B15" ).Value <> Empty Then
7             Dim VendRow As Long
8             VendRow = Range("B15" ).Value 'Vendor Row
9             Range("L7" ).Value = Vendors.Range("C" & VendRow).Value 'Vendor Type
10            Range("J9" ).Value = Vendors.Range("D" & VendRow).Value 'Contact Name
11            Range("L9" ).Value = Vendors.Range("J" & VendRow).Value 'Cell Phone
12            Range("J11" ).Value = Vendors.Range("I" & VendRow).Value 'Email
13        End If
14    End Sub
15
16 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
17     If Target.CountLarge > 1 Then Exit Sub
18
19     'On Project Selection, Load Jobs
20     If Not Intersect(Target, Range("D6:D99" )) Is Nothing And Range("D" & Target.Row).
21     Value <> Empty Then
22         Range("B1" ).Value = Target.Row 'Selected Project Row
23         Range("B2" ).Value = Range("C" & Target.Row).Value 'Set Project ID
24         Project_ListJobs 'Run Macro To load Jobs
25         Schedule_Refresh 'Update Schedule
26     End If
27
28     'On Job Selection
29     If Not Intersect(Target, Range("F5:G999" )) Is Nothing And Range("F" & Target.Row).
30     Value <> Empty Then
31         Range("B6" ).Value = Target.Row 'Selected Row
32         Range("B7" ).Value = Range("E" & Target.Row).Value 'Selected Job ID
33         Job_Load 'Run Macro To Load Job
34     End If
35
36     'On Attachment Selection
37     If Not Intersect(Target, Range("I29:J38" )) Is Nothing And Range("H" & Target.Row).
38     Value <> Empty Then
39         Range("B16" ).Value = Target.Row 'Set selected Row
40         Attach_DisplayThumbnail 'Run Macro to display thumbnail
41     End If
42 End Sub
```

1 Option Explicit
2

1 Option Explicit

2

```
1 Option Explicit
2
3
4
5 Sub Admin_SetStatusTypeColor()
6     ActiveCell.Interior.Color = Admin.Shapes(Application.Caller).Fill.ForeColor.RGB
7     Admin.Shapes("ColorPalette").Visible = msoFalse
8 End Sub
9
10 Sub BrowseForAttachFolder()
11     Dim JobFolder As FileDialog
12     Set JobFolder = Application.FileDialog(msoFileDialogFolderPicker)
13     With JobFolder
14         .Title = "Browse For JobAttachments Folder"
15         .AllowMultiSelect = False
16         If .Show <> -1 Then GoTo NoSelection
17         Admin.Range("D3").Value = .SelectedItems(1)
18     End With
19     NoSelection:
20 End Sub
```

```

1 Option Explicit
2 Dim AttachFile As FileDialog, AttFolder As String, FilePath As String, FileName As String
3 Dim JobID As String, FileType As String, AttName As String
4 Dim LastRow As Long, LastResultRow As Long, AttachRow As Long, SelRow As Long, ResultRow As Long
5
6 Sub Attach_Add()
7     With RenPlan
8         If .Range("B3").Value = Empty Then 'Save Job before adding attachments
9             MsgBox "Please save this Job before adding attachments"
10            Exit Sub
11        End If
12
13        If [AttachFolder] = "" Then
14            MsgBox "Please select an attachment folder in the Admin screen"
15            Exit Sub
16        End If
17
18        JobID = .Range("B7").Value 'Job ID
19        Set AttachFile = Application.FileDialog(msoFileDialogFilePicker)
20        With AttachFile
21            .Title = "Select file to attach"
22            .Filters.Add "All Files", "*.*"
23            If .Show <> -1 Then GoTo NoSelection
24            FilePath = .SelectedItems(1) 'Get Full File Path
25            FileName = Dir(FilePath) 'Get File Name
26            FileType = Right(FileName, Len(FileName) - InStrRev(FileName, ".")) 'Get File Type (File Extension)
27            If FilePath = [AttachFolder] & "\" & FileName Then GoTo FileInDest
28            FileCopy FilePath, [AttachFolder] & "\" & FileName 'Copy To attachments folder
29            FileInDest:
30        End With
31        With JobAttach
32            AttachRow = .Range("A9999").End(xlUp).Row + 1 'First Avail. Attachment Row
33            .Range("A" & AttachRow).Value = JobID 'Jobining ID
34            .Range("B" & AttachRow).Value = RenPlan.Range("J5").Value 'Job Name
35            .Range("C" & AttachRow).Value = Date 'Current Added On Date
36            .Range("D" & AttachRow).Value = FileName 'File Name
37            .Range("E" & AttachRow).Value = FileType 'File Type
38            .Range("F" & AttachRow).Value = "=Row()" 'DB Row
39        End With
40
41        Attach_Refresh 'Refresh List To Display New Attachment
42        .Range("I29").Select 'Select First Item to Display Thumbnail (if any)
43        NoSelection:
44    End With
45 End Sub
46
47 Sub Attach_Delete()
48     If MsgBox("Are you sure you want to delete this attachment?", vbYesNo, "Delete Attachment") = vbNo Then Exit Sub
49     With RenPlan
50         If .Range("B16").Value = Empty Then
51             MsgBox "Please select an attachment to delete"
52             Exit Sub
53         End If
54         SelRow = .Range("B16").Value 'Selected Row
55         If RenPlan.Range("H" & SelRow).Value = Empty Then
56             MsgBox "Please select an Attachment to delete first"

```

1 2 3

```

57     Exit Sub
58 End If
59
60 On Error Resume Next
61 RenPlan.Shapes("AttachThumb").Delete 'Delete Thumbnail if any
62 On Error GoTo 0
63
64 SelRow = .Range("B16").Value 'Selected Row
65 AttachRow = .Range("H" & SelRow).Value 'Attachment DB Row
66 JobAttach.Range(AttachRow & ":" & AttachRow).EntireRow.Delete
67 Attach_Refresh
68 RenPlan.Range("G29").Select
69 Attach_DisplayThumbnail
70 End With
71 End Sub
72
73
74
75
76 Sub Attach_DisplayThumbnail()
77     With RenPlan
78         On Error Resume Next
79         .Shapes("AttachThumb").Delete 'Clear any existing attachment file
80         On Error GoTo 0
81         If .Range("B16").Value = Empty Then Exit Sub 'Exit on No AttachmentRow
82         SelRow = .Range("B16").Value 'Set Attachment Row
83         AttachRow = .Range("H" & SelRow).Value 'Attach DB Row
84         FileName = JobAttach.Range("D" & AttachRow).Value 'File Name
85         FileType = JobAttach.Range("E" & AttachRow).Value 'File Type
86         FilePath = [AttachFolder] & "\" & FileName 'Full Attachment File Path
87         If FileType = "jpg" Or FileType = "png" Or FileType = "gif" Or FileType =
            "jpeg" Or FileType = "bmp" Then 'Picture Types
88             On Error GoTo NoAttach
89             If Dir(FilePath, vbDirectory) = "" Then Exit Sub
90             .Pictures.Insert(FilePath).Name = "AttachThumb"
91             With .Shapes("AttachThumb")
92                 .Visible = msoCTrue
93                 .LockAspectRatio = True
94                 If .Width > .Height Then .Width = 120 Else: .Height = 110
95                 .Left = RenPlan.Range("K29").Left + (RenPlan.Range("K:L").Width - .Width)
96                 .Top = RenPlan.Range("K29").Top + (RenPlan.Range("29:38").Height - .
                    Height) / 2
97             End With
98         End If
99     End With
100     NoAttach:
101 End Sub
102 Sub Attach_Open()
103     With RenPlan
104         If .Range("B16").Value = Empty Then
105             MsgBox "Please select an attachment to open"
106             Exit Sub
107         End If
108         AttachRow = .Range("B16").Value 'Selected Attachment Row
109         FilePath = [AttachFolder] & "\" & .Range("J" & AttachRow).Value 'Attachment
            File Path
110         If Dir(FilePath) = "" Or FilePath = "" Then Exit Sub
111         ThisWorkbook.FollowHyperlink(FilePath) 'Open Selected File
112     End With
113 End Sub

```

```
114
115 Sub Attach_Refresh()
116     If RenPlan.Range("B7" ).Value = "" Then Exit Sub 'Job ID
117     RenPlan.Range("B16,H29:J38" ).ClearContents 'Clear Old Attachemets fields
118     On Error Resume Next
119     RenPlan.Shapes("AttachThumb" ).Delete 'Delete any existing attachment
120     On Error GoTo 0
121     AttachRow = 29 'Set Starting Attachments Row
122     With JobAttach 'With Attachments sheet
123         LastRow = .Range("A9999" ).End(xlUp).Row
124         If LastRow < 4 Then Exit Sub
125         .Range("A3:F" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
126             "K2:K3" ), CopyToRange:=.Range("N2:P2" ), Unique:=True
127         LastResultRow = .Range("N9999" ).End(xlUp).Row 'Last Filtered Row Results
128         If LastResultRow < 3 Then Exit Sub 'If no Attach. found then attachments
129         For ResultRow = LastResultRow To 3 Step -1 'Use Reverse For next to put most
130             recent Attach. First
131             RenPlan.Range("H" & AttachRow & ":J" & AttachRow).Value = .Range("N" &
132                 ResultRow & ":P" & ResultRow).Value 'Copy Over Attachment Info
133             AttachRow = AttachRow + 1
134             If AttachRow = 39 Then Exit Sub 'Exit If Table is Full
135         Next ResultRow
136     End With
137 End Sub
```



```

1 Option Explicit
2
3 Dim LastRow As Long, LastResultRow As Long, JobRow As Long, JobCol As Long, SelRow As
  Long, ProjRow As Long
4
5
6 Sub Job_Delete()
7   With RenPlan
8     If MsgBox("Are you sure you want to delete this Job?" , vbYesNo, "Delete Job" ) =
        vbNo Then Exit Sub
9     If .Range("B8" ).Value = Empty Then GoTo NotSaved
10    JobRow = .Range("B8" ).Value 'Job Row
11    LastRow = ProjJobs.Range("A99999" ).End(xlUp).Row
12    ProjJobs.Range("A" & JobRow & ":M" & JobRow).ClearContents 'Clear Range
13    If LastRow < 4 Then GoTo NotSaved 'No Sorting on a single record
14    With ProjJobs.Sort
15      .SortFields.Clear
16      .SortFields.Add Key:=ProjJobs.Range("A3" ), SortOn:=xlSortOnValues, Order:=
        xlAscending, DataOption:=xlSortNormal 'Sort
17      .SetRange ProjJobs.Range("A4:M" & LastRow) 'Set Range
18      .Apply 'Apply Sort
19    End With
20    NotSaved:
21    Job_New 'Run macro to create new Job
22    Project_ListJobs 'Update Job list
23    Schedule_Refresh
24  End With
25 End Sub
26
27
28
29 Sub Job_Load()
30   With RenPlan
31     .Range("B16,J5,L5,J7,L7,J9,L9,J11:L11,J13,L13,J15,L15,J17,L17,J19:L23,H29:J38" ).
        ClearContents 'Clear Job Fields
32     If .Range("B8" ).Value = Empty Then
33       MsgBox "Please select a correct Job to load"
34       Exit Sub
35     End If
36     JobRow = .Range("B8" ).Value 'Job Row
37     .Range("B5" ).Value = True 'Set Job Load To True
38     For JobCol = 3 To 12
39       .Range(ProjJobs.Cells(1, JobCol).Value).Value = ProjJobs.Cells(JobRow, JobCol)
        .Value 'Bring over data
40     Next JobCol
41     RenPlan.Range("B5" ).Value = False 'Set Job Load To False
42   End With
43   Attach_Refresh 'Load Attachments, if any
44 End Sub
45
46 Sub Job_New()
47   RenPlan.Range(
48     "B7,B6,B7,B16,J5,L5,J7,L7,J9,L9,J11:L11,J13,L13,J15,L15,J17,L17,J19:L23,H29:J38" ).
        ClearContents 'Clear Job Fields
49   On Error Resume Next
50   RenPlan.Shapes("AttachThumb" ).Delete
51   On Error GoTo 0
52 End Sub
53
54 Sub Job_SaveUpdate()
55   With RenPlan

```

```

1 2
55 If .Range("J5" ).Value = Empty Then 'No job name
56     MsgBox "Please make sure to enter a Job Name before saving"
57     Exit Sub
58 End If
59 If .Range("B8" ).Value = Empty Then 'New Job
60     JobRow = ProjJobs.Range("A9999" ).End(xlUp).Row + 1           'First Avail Row
61     .Range("B7" ).Value = .Range("B9" ).Value 'New Job. ID
62     ProjJobs.Range("A" & JobRow).Value = .Range("B7" ).Value 'Set Job ID
63 Else 'Existing Job
64     JobRow = .Range("B8" ).Value 'Existing Job Row
65 End If
66 For JobCol = 2 To 13
67     ProjJobs.Cells(JobRow, JobCol).Value = .Range(ProjJobs.Cells(1, JobCol).Value)
        .Value 'Save/Update Data
68 Next JobCol
69 Project_ListJobs 'Update Job list
70 Schedule_Refresh 'Update Schedule
71 MsgBox "Job has been saved"
72 End With
73 End Sub
74 Sub Project_AddNew()
75     Dim ProjName As String
76     ProjName = InputBox("Please enter Project Name" , "New Project" )
77     If ProjName = "" Then Exit Sub
78     ProjRow = Projects.Range("A9999" ).End(xlUp).Row + 1
79     RenPlan.Range("B2" ).Value = RenPlan.Range("B4" ).Value 'Next Project ID
80     Projects.Range("A" & ProjRow).Value = RenPlan.Range("B4" ).Value 'Next Project ID
81     Projects.Range("B" & ProjRow).Value = ProjName 'Project Name
82     Projects_LoadProjects 'Load Project List
83     Project_ListJobs 'Run Macro to clear/List jobs
84     Job_New 'Run Macro To Add New Job
85 End Sub
86 Sub Project_ListJobs()
87     RenPlan.Range("B6,E5:G9999" ).ClearContents 'Clear existing Jobs
88     With ProjJobs
89         LastRow = .Range("A9999" ).End(xlUp).Row 'Last Row of data
90         If LastRow < 4 Then Exit Sub
91         .Range("A3:J" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
            "Q2:Q3" ), CopyToRange:=.Range("S2:U2" ), Unique:=True
92         LastResultRow = .Range("S9999" ).End(xlUp).Row 'Last Result Row
93         If LastResultRow < 3 Then Exit Sub
94         If LastResultRow < 4 Then GoTo SkipSort
95         'Sort Based On Job Name
96         With .Sort
97             .SortFields.Clear
98             .SortFields.Add Key:=ProjJobs.Range("S3" ), SortOn:=xlSortOnValues, Order:=
                xlAscending, DataOption:=xlSortNormal 'Sort
99             .SetRange ProjJobs.Range("S3:U" & LastResultRow) 'Set Range
100            .Apply 'Apply Sort
101        End With
102        SkipSort:
103    End With
104    'Add Project Jobs & Selected Row
105    With RenPlan
106        .Range("E5:G" & LastResultRow + 2).Value = ProjJobs.Range("S3:U" &
            LastResultRow).Value 'Bring Over Jobs
107        On Error Resume Next
108        SelRow = .Range("E5:E9999" ).Find(.Range("B7" ).Value, , xlFormulas, xlWhole).Row
109        If SelRow <> 0 And .Range("B7" ).Value <> Empty Then .Range("B6" ).Value = SelRow
            'Set Selected Job Row
    End With
1 2

```

```
1 2
110 On Error GoTo 0
111 End With
112 End Sub
113 Sub Projects_LoadProjects()
114 RenPlan.Range("B1,C6:D99999").ClearContents 'Clear Existing Projects
115 With Projects
116 LastRow = .Range("A99999").End(xlUp).Row
117 If LastRow < 4 Then Exit Sub
118 .Range("G3:H" & LastRow - 1).Value = .Range("A4:B" & LastRow).Value 'Bring
Over Data
119 If LastRow < 5 Then GoTo NoSort
120 'Sort Alphabetally
121 With .Sort
122 .SortFields.Clear
123 .SortFields.Add Key:=Projects.Range("H3"), SortOn:=xlSortOnValues, Order:=
xlAscending, DataOption:=xlSortNormal 'Sort
124 .SetRange Projects.Range("G3:H" & LastRow - 1) 'Set Range
125 .Apply 'Apply Sort
126 End With
127 NoSort:
128 RenPlan.Range("C6:D" & LastRow + 2).Value = .Range("G3:H" & LastRow - 1).Value
'Bring Over Projects
129 End With
130 With RenPlan
131 On Error Resume Next
132 SelRow = .Range("C6:C9999").Find(.Range("B2").Value, , xlFormulas, xlWhole).Row
133 If SelRow <> 0 And .Range("B2").Value <> Empty Then .Range("B1").Value = SelRow
'Set Selected Project Row
134 On Error GoTo 0
135 End With
136 End Sub
```

```

1 Option Explicit
2 Dim JobShp As Shape
3 Dim LastRow As Long, ResultRow As Long, LastResultRow As Long, JobDays As Long
4 Dim ScRow As Long, ScStartCol As Long, ScEndCol As Long, StatusRow As Long, CompPerc As
  Double
5 Dim JobID As String, JobName As String, JobStatus As String, JobType As String,
  StatusColor As String, VendName As String
6 Dim ScStartDt As Date, ScEndDt As Date, StartDt As Date, EndDt As Date
7
8 Sub Sched_JobSelect()
9     JobID = Replace(Application.Caller, Left(Application.Caller, 6), "")
10    RenPlan.Range("B7").Value = JobID
11    Job_Load
12 End Sub
13
14 Sub Sched_NextMonth()
15     With RenPlan
16         If .Range("B11").Value = 12 Then 'Increase Year by 1, and set to January
17             If .Range("B13").Value = 7 Then
18                 MsgBox "You are on the last available year. Please update the years in the
  Admin screen"
19                 Exit Sub
20             End If
21             .Range("B13").Value = .Range("B13").Value + 1 'Increase year
22             .Range("B11").Value = 1 'Set to January
23         Else: 'Non December
24             .Range("B11").Value = .Range("B11").Value + 1 'Increase Month
25         End If
26     End With
27     Schedule_Refresh 'Refresh Schedule
28 End Sub
29
30
31 Sub Sched_PrevMonth()
32     With RenPlan
33         If .Range("B11").Value = 1 Then 'Reduce Year by 1, and set to December
34             If .Range("B13").Value = 1 Then
35                 MsgBox "You are on the first available year"
36                 Exit Sub
37             End If
38             .Range("B13").Value = .Range("B13").Value - 1 'Reduce year
39             .Range("B11").Value = 12 'Set to December
40         Else: 'Non -January
41             .Range("B11").Value = .Range("B11").Value - 1 'Reduce Month
42         End If
43     End With
44     Schedule_Refresh 'Refresh Schedule
45 End Sub
46
47 Sub Sched_ThisMonth()
48     With RenPlan
49         .Range("B11").Value = Month(Date) 'Set Current Month
50         On Error Resume Next
51         .Range("B13").Value = Admin.Range("Years").Find(Year(Date), , xlValues, xlWhole
  ).Row - 7 'Set Year #
52         On Error GoTo 0
53     End With
54     Schedule_Refresh 'Refresh Schedule
55 End Sub
56
57 Sub Schedule_Refresh()
58     With RenPlan

```

```

1 2
58 'Clear all Schedule Shapes & Jobs
59 .Range("N9:N99").ClearContents
60 For Each JobShp In .Shapes
61     On Error Resume Next
62     If InStr(JobShp.Name, "RenJob") > 0 Then JobShp.Delete
63     On Error GoTo 0
64 Next JobShp
65 ScStartDt = .Range("O8").Value 'Schedule Start Date (1st Day of Month)
66 ScEndDt = .Range("B14").Value 'Schedule End Date (Last Day of Month)
67 End With
68
69 With ProjJobs
70     .Range("AB3:AN9999").ClearContents 'Clear Prev. Results
71     LastRow = .Range("A9999").End(xlUp).Row
72     If LastRow < 4 Then Exit Sub
73     .Range("A3:M" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
74         "W2:AA5"), CopyToRange:=.Range("AC2:AN2"), Unique:=True
75     LastResultRow = .Range("AC9999").End(xlUp).Row
76     If LastResultRow < 3 Then Exit Sub
77     If LastResultRow < 4 Then GoTo SkipSort
78     'Sort Based on Start Date
79     With .Sort
80         .SortFields.Clear
81         .SortFields.Add Key:=ProjJobs.Range("AI3"), SortOn:=xlSortOnValues, Order:=
82             xlAscending, DataOption:=xlSortNormal 'Sort
83         .SetRange ProjJobs.Range("AC3:AN" & LastResultRow) 'Set Range
84         .Apply 'Apply Sort
85     End With
86     SkipSort:
87     RenPlan.Range("N9:N" & LastResultRow + 6).Value = .Range("AD3:AD" &
88         LastResultRow).Value 'Bring Over Job Names
89     For ResultRow = 3 To LastResultRow
90         'Get Variables
91         JobID = .Range("AC" & ResultRow).Value 'Job ID
92         JobName = .Range("AD" & ResultRow).Value 'Job Name
93         JobType = .Range("AE" & ResultRow).Value 'Job Type
94         VendName = .Range("AF" & ResultRow).Value 'Vendor Name
95         StartDt = .Range("AI" & ResultRow).Value 'Start Date
96         JobDays = .Range("AJ" & ResultRow).Value 'Job Days
97         JobStatus = .Range("AK" & ResultRow).Value 'Job Status
98         CompPerc = .Range("AL" & ResultRow).Value 'Job Completion %
99         EndDt = .Range("AN" & ResultRow).Value 'End Date
100
101         'Set Starting Column
102         If StartDt <= ScStartDt Then 'Set To first Column
103             ScStartCol = 15
104         Else 'Starts Within the Month
105             ScStartCol = 15 + (StartDt - ScStartDt) 'Set Start Column
106         End If
107
108         'Set Ending Column
109         If EndDt >= ScEndDt Then 'End on or after last day of the month - Set To
110             last column
111             ScEndCol = 15 + (ScEndDt - ScStartDt)
112         Else 'Ends Within the Month
113             ScEndCol = 14 + (EndDt - ScStartDt) 'Set End Column
114         End If
115         ScRow = ResultRow + 6 'Determine Scheduld Row
116
117         'Determine Job Status Color
118         On Error Resume Next

```

1 2 3

```

115   1 2 3   StatusRow = Admin.Range("JobStatus").Find(JobStatus, , xlValues, xlWhole).Row
116   On Error GoTo 0
117   If StatusRow <> 0 Then StatusColor = Admin.Range("F" & StatusRow).Interior.
    Color 'Get Interior cell color
118
119   With RenPlan
120     'Create Job Shape
121     .Shapes("SampleJobShp").Duplicate.Name = "RenJob" & JobID 'Unique Name
122     With .Shapes("RenJob" & JobID)
123       If StatusColor <> "" Then .Fill.ForeColor.RGB = StatusColor 'Set Shape
        Job Color
124       With .Fill 'Set Gradient and color based on completion %
125         .ForeColor.RGB = StatusColor 'Set Status Color
126         .TwoColorGradient msoGradientVertical, 1
127         .GradientStops.Insert RGB(255, 255, 255), CompPerc, , 3 'White Color
        Backing
128         .GradientStops(2).Color = StatusColor 'Set Status Color
129         .GradientStops(2).Position = CompPerc
130       End With
131       .Left = RenPlan.Cells(ScRow, ScStartCol).Left
132       .Top = RenPlan.Cells(ScRow, ScStartCol).Top
133       .Width = Range(RenPlan.Cells(ScRow, ScStartCol), RenPlan.Cells(ScRow,
        ScEndCol)).Width
134       .TextFrame2.TextRange.Text = VendName 'Set Vendor Name
135       .Line.ForeColor.RGB = StatusColor 'Set Border Color
136       .OnAction = "Sched_JobSelect"
137     End With
138   End With
139   NextJob:
140   Next ResultRow
141 End With
142 End Sub

```

A

ActiveCell, 9
 Add, 10, 13-15, 17
 Admin, 9, 16, 18
 Admin_SetStatusTypeColor, 9
 AdvancedFilter, 12, 14, 17
 AllowMultiSelect, 9
 Application, 9, 10, 16
 Apply, 13-15, 17
 Attach_Add, 10
 Attach_Delete, 10
 Attach_DisplayThumbnail, 6, 11
 Attach_Open, 11
 Attach_Refresh, 10-13
 AttachFile, 10
 AttachFolder, 10, 11
 AttachRow, 10-12
 AttFolder, 10
 AttName, 10

B

BrowseForAttachFolder, 9

C

Caller, 9, 16
 Cells, 13, 14, 18
 Clear, 13-15, 17
 ClearContents, 12-15, 17
 Color, 9, 18
 CompPerc, 16-18
 CopyToRange, 12, 14, 17
 CountLarge, 2, 6
 CriteriaRange, 12, 14, 17

D

DataOption, 13-15, 17
 Delete, 11-13, 17
 Dir, 10, 11
 Duplicate, 18

E

Empty, 2, 6, 10, 11, 13-15
 EndDt, 16, 17
 EntireRow, 11
 Explicit, 2-10, 13, 16

F

FileCopy, 10
 FileDialog, 9, 10
 FileInDest, 10
 FileName, 10, 11
 FilePath, 10, 11
 FileType, 10, 11
 Fill, 9, 18
 Filters, 10
 Find, 14-16, 18
 FollowHyperlink, 11
 ForeColor, 9, 18

G

GradientStops, 18

H

Height, 11

I

InputBox, 14
 Insert, 11, 18
 InStr, 17

InStrRev, 10
 Interior, 9, 18
 Intersect, 2, 6

J

Job_Delete, 13
 Job_Load, 6, 13, 16
 Job_New, 13, 14
 Job_SaveUpdate, 13
 JobAttach, 10-12
 JobCol, 13, 14
 JobDays, 16, 17
 JobFolder, 9
 JobID, 10, 16-18
 JobName, 16, 17
 JobRow, 13, 14
 JobShp, 16, 17
 JobStatus, 16-18
 JobType, 16, 17

K

Key, 13-15, 17

L

LastResultRow, 10, 12-14, 16, 17
 LastRow, 10, 12-17
 Left, 2, 11, 16, 18
 Len, 10
 Line, 18
 LockAspectRatio, 11

M

Month, 16
 MsgBox, 10, 11, 13, 14, 16
 msoCTrue, 2, 11
 msoFalse, 2, 9
 msoFileDialogFilePicker, 10
 msoFileDialogFolderPicker, 9
 msoGradientVertical, 18

N

Name, 11, 17, 18
 NextJob, 18
 NoAttach, 11
 NoSelection, 9, 10
 NoSort, 15
 NotSaved, 13

O

OnAction, 18
 Order, 13-15, 17

P

Pictures, 11
 Position, 18
 Project_AddNew, 14
 Project_ListJobs, 6, 13, 14
 Projects, 14, 15
 Projects_LoadProjects, 14, 15
 ProjJobs, 13, 14, 17
 ProjName, 14
 ProjRow, 13, 14

R

Range, 2, 6, 9-18
 RenPlan, 10-18
 Replace, 16
 ResultRow, 10, 12, 16-18
 RGB, 9, 18
 Right, 10

Row, 2, 6, 10, 12-18

S

ScEndCol, 16-18
 ScEndDt, 16, 17
 Sched_JobSelect, 16
 Sched_NextMonth, 16
 Sched_PrevMonth, 16
 Sched_ThisMonth, 16
 Schedule_Refresh, 6, 13, 14, 16
 ScRow, 16-18
 ScStartCol, 16-18
 ScStartDt, 16, 17
 SelectedItems, 9, 10
 SelRow, 10, 11, 13-15
 SetRange, 13-15, 17
 Shape, 16
 Shapes, 2, 9, 11-13, 17, 18
 Show, 9, 10
 SkipSort, 14, 17
 Sort, 13-15, 17
 SortFields, 13-15, 17
 SortOn, 13-15, 17
 StartDt, 16, 17
 StatusColor, 16, 18
 StatusRow, 16, 18

T

Target, 2, 6
 Text, 18
 TextFrame2, 18
 TextRange, 18
 ThisWorkbook, 11
 Title, 9, 10
 Top, 2, 11, 18
 TwoColorGradient, 18

U

Unique, 12, 14, 17

V

Value, 2, 6, 9-17
 vbDirectory, 11
 vbNo, 10, 13
 vbYesNo, 10, 13
 VendName, 16-18
 Vendors, 6
 VendRow, 6
 Visible, 2, 9, 11

W

Width, 11, 18
 Worksheet_Change, 6
 Worksheet_SelectionChange, 2, 6

X

xlAscending, 13-15, 17
 xlFilterCopy, 12, 14, 17
 xlFormulas, 14, 15
 xlSortNormal, 13-15, 17
 xlSortOnValues, 13-15, 17
 xlUp, 10, 12-15, 17
 xlValues, 16, 18
 xlWhole, 14-16, 18

Y

Year, 16

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