

VBA SOURCE CODE BOOK



How To Create Your Own Collectibles Tracking 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](#)

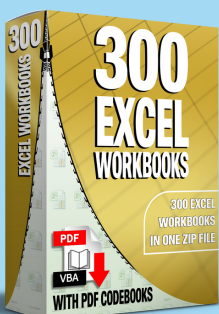
-



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](#)

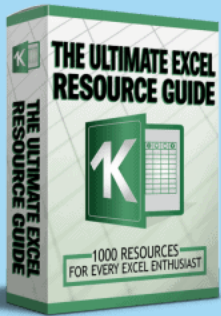
-



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](#)

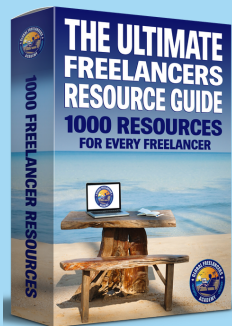
OUR COURSES & PRODUCTS



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_Change [Sub].....	2
AttachDB.....	3
(Declarations).....	3
CollDB.....	4
(Declarations).....	4
CollSht.....	5
(Declarations).....	5
Worksheet_Change [Sub].....	5
Worksheet_SelectionChange [Sub].....	5
Sheet1.....	6
(Declarations).....	6
ThisWorkbook.....	7
(Declarations).....	7
Forms.....	8
AttachFrm.....	8
(Declarations).....	8
CancelBtn_Click [Sub].....	8
SaveBtn_Click [Sub].....	8
Modules.....	9
Admin_Macros.....	9
(Declarations).....	9
BrowseForPicFolder [Sub].....	9
Attach_Macros.....	10
(Declarations).....	10
Attach_Add [Sub].....	10
Attach_Delete [Sub].....	10
Attach_DisplayPicture [Sub].....	11
Attach_Open [Sub].....	11
Attach_Refresh [Sub].....	11
Attach_Save [Sub].....	12
CollItem_Macros.....	13
(Declarations).....	13
Item_AddPicture [Sub].....	13
Item_AddTag [Sub].....	13
Item_ClearFilter [Sub].....	13
Item_Delete [Sub].....	13
Item_DisplayPicture [Sub].....	14
Item_FilterByTag [Sub].....	14
Item_ListItems [Sub].....	14
Item_Load [Sub].....	15
Item_LoadTags [Sub].....	15
Item_New [Sub].....	16
Item_RemoveTag [Sub].....	16
Item_SaveUpdate [Sub].....	17
Item_Select [Sub].....	17
ItemClearPicture [Sub].....	17
ItemPrintVisibleItems [Sub].....	17

```
1 Option Explicit
2
3
4 Private Sub Worksheet_Change(ByVal Target As Range)
5     'On Custom Field Type Change, Update Format
6     If Not Intersect(Target, Range("G7:G12" )) Is Nothing Then
7         Dim FieldType As String
8         FieldType = Target.Value
9         Select Case FieldType
10            Case Is = "Text"
11                Range("H" & Target.Row).Value = "@"
12            Case Is = "Amount"
13                Range("H" & Target.Row).Value = "$#,##0.00"
14            Case Is = "Percentage"
15                Range("H" & Target.Row).Value = "%"
16            Case Is = "Date"
17                Range("H" & Target.Row).Value = "m/d/yyyy"
18            Case Else
19                Range("H" & Target.Row).Value = "@"
20            End Select
21        End If
22
23        'On Custom Field Type Change, Update Format
24        If Not Intersect(Target, Range("H7:H12" )) Is Nothing Then
25            Dim RowNumb As Long
26            RowNumb = Target.Row
27            Select Case RowNumb
28            Case Is = 7
29                CollSht.Range("J19" ).NumberFormat = Target.Value
30            Case Is = 8
31                CollSht.Range("M19" ).NumberFormat = Target.Value
32            Case Is = 9
33                CollSht.Range("J21" ).NumberFormat = Target.Value
34            Case Is = 10
35                CollSht.Range("M21" ).NumberFormat = Target.Value
36            Case Is = 11
37                CollSht.Range("J23" ).NumberFormat = Target.Value
38            Case Is = 12
39                CollSht.Range("M23" ).NumberFormat = Target.Value
40            End Select
41        End If
42    End Sub
43
```

1 Option Explicit

2

1
2

Option Explicit


```
1 Option Explicit
2
3 Private Sub Worksheet_Change(ByVal Target As Range)
4     If Not Intersect(Target, Range("G4,E4" )) Is Nothing Then
5         If Range("E4" ).Value <> "" And Range("G4" ).Value = Empty Then Exit Sub
6         If Range("G4" ).Value <> "" And Range("E4" ).Value = Empty Then Exit Sub
7         Application.EnableEvents = False
8         Item_ListItems
9         Application.EnableEvents = True
10    End If
11 End Sub
12
13 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
14     If Target.CountLarge > 1 Then Exit Sub
15     If Shapes("DelAttBtn" ).Visible = True Then Shapes("DelAttBtn" ).Visible = msoFalse
16     'Hide button if visible
17     If Not Intersect(Target, Range("O5:Q9999" )) Is Nothing And Range("O" & Target.Row)
18     .Value <> Empty Then
19         Range("B6" ).Value = Target.Row
20         With Shapes("DelAttBtn" ) 'Set Delete Button
21             .Left = Range("R" & Target.Row).Left
22             .Top = Range("R" & Target.Row).Top
23             .Visible = msoCTrue
24         End With
25     Attach_DisplayPicture
26 End If
27 End Sub
```

1
2

Option Explicit

1 Option Explicit

2

```
1 Option Explicit
2
3 Private Sub CancelBtn_Click()
4     Unload Me
5 End Sub
6
7 Private Sub SaveBtn_Click()
8     Attach_Save
9 End Sub
```

```
1 Option Explicit
2
3 Sub BrowseForPicFolder()
4     Dim PicFolder As FileDialog
5     Set PicFolder = Application.FileDialog(msoFileDialogFolderPicker)
6     With PicFolder
7         .Title = "Browse Picture & Attachments Folder"
8         .AllowMultiSelect = False
9         If .Show <> -1 Then GoTo NoSelection
10        Admin.Range("D3" ).Value = .SelectedItems(1)
11    End With
12    NoSelection:
13 End Sub
14
15
16
```

```

1 Option Explicit
2 Dim AttachFile As FileDialog, PicFolder As String, PicPath As String, FileName As String, PicName As String
3 Dim ItemID As String, FileType As String, AttachName As String, FileExt As String, AttachDesc As String, ItemName 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 CollSht
8         If .Range("B3").Value = Empty Then 'Save Item before adding attachments
9             MsgBox "Please save this Item before adding attachments"
10            Exit Sub
11        End If
12
13        If [PictureFolder] = "" Then
14            MsgBox "Please select an Picture/Attachment folder in the Admin screen"
15            Exit Sub
16        End If
17        PicFolder = [PictureFolder] 'Set Folder
18        ItemID = .Range("B2").Value 'Item ID
19        Set AttachFile = Application.FileDialog(msoFileDialogFilePicker)
20        With AttachFile
21            .Title = "Select File To Attach"
22            On Error Resume Next
23            .Filters.Delete
24            On Error GoTo 0
25            .Filters.Add "All Files", "*.*"
26            If .Show <> -1 Then GoTo NoSelection
27            FileName = Dir(.SelectedItems(1)) 'Get File Name
28            AttachFrm.FileName = FileName 'Set File Name
29            If .SelectedItems(1) = PicFolder & "\" & FileName Then GoTo SkipCopy
30            FileCopy .SelectedItems(1), PicFolder & "\" & FileName 'Copy To Attachments folder
31            SkipCopy:
32            AttachFrm.Show
33        End With
34        NoSelection:
35    End With
36 End Sub
37
38 Sub Attach_Delete()
39     If MsgBox("Are you sure you want to delete this attachment?", vbYesNo, "Delete Attachment") = vbNo Then Exit Sub
40     With CollSht
41         If .Range("B6").Value = Empty Then
42             MsgBox "Please select an attachment to delete"
43             Exit Sub
44         End If
45         SelRow = .Range("B6").Value 'Selected Row
46         If CollSht.Range("R" & SelRow).Value = Empty Then
47             MsgBox "Please select an Attachment to delete first"
48             Exit Sub
49         End If
50         CollSht.Shapes("ItemPic").Visible = msoFalse
51         AttachRow = .Range("R" & SelRow).Value 'Attachment DB Row
52         AttachDB.Range(AttachRow & ":" & AttachRow).EntireRow.Delete
53         Attach_Refresh
54         CollSht.Range("O5").Select
55     End With
56 End Sub

```

```

57
58
59
60
61
62 Sub Attach_DisplayPicture()
63     With CollSht
64         .Shapes("ItemPic" ).Visible = msoFalse
65         If .Range("B6" ).Value = Empty Then Exit Sub 'Exit on no selected row
66         SelRow = .Range("B6" ).Value 'Selected Row
67         AttachRow = .Range("R" & SelRow).Value 'Attachment Row
68         PicName = AttachDB.Range("C" & AttachRow).Value 'Picture Name
69         FileExt = AttachDB.Range("G" & AttachRow).Value 'File Extension
70         If FileExt = "png" Or FileExt = "jpg" Or FileExt = "jpeg" Or FileExt = "bmp"
71             Then 'Display Picture
72                 PicFolder = [PictureFolder]
73                 If PicName = Empty Or [PictureFolder] = Empty Then Exit Sub
74                 PicPath = PicFolder & "\" & PicName
75                 If Dir(PicPath, vbDirectory) = "" Then Exit Sub
76                 .Shapes("ItemPic" ).Fill.UserPicture PicPath
77                 .Shapes("ItemPic" ).Visible = msoCTrue
78             End If
79         End With
80     End Sub
81 Sub Attach_Open()
82     With CollSht
83         If .Range("B6" ).Value = Empty Then
84             MsgBox "Please select an attachment to open"
85             Exit Sub
86         End If
87         SelRow = .Range("B6" ).Value
88         AttachRow = .Range("R" & SelRow).Value 'Selected Attachment Row
89         PicPath = [PictureFolder] & "\" & AttachDB.Range("C" & AttachRow).Value
90         'Attachment File Path
91         If Dir(PicPath) = "" Or PicPath = "" Then Exit Sub
92         ThisWorkbook.FollowHyperlink (PicPath) 'Open Selected File
93     End With
94 End Sub
95 Sub Attach_Refresh()
96     If CollSht.Range("B3" ).Value = "" Then Exit Sub 'Exit on missing id Item ID
97     CollSht.Range("B6,05:R999" ).ClearContents 'Clear Old Attachemets fields
98     CollSht.Shapes("ItemPic" ).Visible = msoFalse 'Hide Picture for now
99     With AttachDB 'With Attachments sheet
100         LastRow = .Range("A9999" ).End(xlUp).Row
101         If LastRow < 4 Then Exit Sub
102         .Range("A3:H" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
103             "K2:K3" ), CopyToRange:=.Range("N2:Q2" ), Unique:=True
104         LastResultRow = .Range("N9999" ).End(xlUp).Row 'Last Filtered Row Results
105         If LastResultRow < 3 Then Exit Sub 'If no Attach. found then attachments
106         If LastResultRow < 4 Then GoTo NoSort
107         'Sort Based on Row (Descending)
108         With .Sort
109             .SortFields.Clear
110             .SortFields.Add Key:=AttachDB.Range("Q3" ), SortOn:=xlSortOnValues, Order:=
111                 xlDescending, DataOption:=xlSortNormal 'Sort
112             .SetRange AttachDB.Range("N3:Q" & LastResultRow) 'Set Range
113             .Apply 'Apply Sort
114         End With
115         NoSort:
116         CollSht.Range("O5:R" & LastResultRow + 2).Value = .Range("N3:Q" & LastResultRow

```

1 2

```
1 2
).Value
114 End With
115 End Sub
116
117 Sub Attach_Save()
118 With AttachFrm
119 If .AttachName.Value = "" Or .FileType.Value = Empty Then
120 MsgBox "Please make sure to add in a File Name and Type Before saving"
121 Exit Sub
122 End If
123
124 ItemID = CollSht.Range("B2").Value 'Item ID
125 ItemName = CollSht.Range("J3").Value ' Item Name
126 FileName = .FileName.Value 'File Name
127 AttachName = .AttachName.Value 'Attachment Name
128 AttachDesc = .Description.Value 'Attachment Description
129 FileType = .FileType.Value 'File type
130 FileExt = Right(FileName, Len(FileName) - InStrRev(FileName, ".")) 'Get File
Type (File Extention)
131 Unload AttachFrm
132 End With
133 With AttachDB
134 AttachRow = .Range("A99999").End(xlUp).Row + 1
135 .Range("A" & AttachRow).Value = ItemID 'Set Item ID
136 .Range("B" & AttachRow).Value = ItemName 'Item Name
137 .Range("C" & AttachRow).Value = FileName 'File Name
138 .Range("D" & AttachRow).Value = AttachName 'Attachment Name
139 .Range("E" & AttachRow).Value = AttachDesc 'Attach. Description
140 .Range("F" & AttachRow).Value = FileType 'File Type
141 .Range("G" & AttachRow).Value = FileExt 'File Extension
142 .Range("H" & AttachRow).Value = "=Row()" 'Set DB Row
143 End With
144 Attach_Refresh
145 On Error Resume Next
146 If CollSht.Range("O5").Value <> Empty Then CollSht.Range("O5").Select
147 On Error GoTo 0
148 End Sub
```



```

1 Option Explicit
2 Dim LastRow As Long, LastResultRow As Long, ResultRow As Long
3 Dim ItemCol As Long, ItemRow As Long, TagNumb As Long
4 Dim ItemShp As Shape
5 Dim ItemID As String, ItemName As String, ItemDesc As String, TagArr() As String
6 Dim PicFolder As String, PicName As String, PicPath As String
7 Dim TopPos As Double, LeftPos As Double
8 Sub Item_AddPicture()
9     Dim ItemPic As FileDialog
10    Set ItemPic = Application.FileDialog(msoFileDialogFilePicker)
11    PicFolder = [PictureFolder]
12    If PicFolder = "" Or Dir(PicFolder, vbDirectory) = "" Then
13        BrowseForPicFolder
14        PicFolder = [PictureFolder]
15        If PicFolder = "" Or Dir(PicFolder, vbDirectory) = "" Then Exit Sub
16    End If
17    With ItemPic
18        .Title = "Select Item Picture"
19        .Filters.Add "Select Item Picture", "*.jpg, *.png, *.jpeg, *.gif", 1
20        If .Show <> -1 Then GoTo NoSelection
21        PicPath = PicFolder & "\" & Dir(.SelectedItems(1))
22        If .SelectedItems(1) = PicPath Then GoTo SkipCopy
23        If Dir(PicPath, vbDirectory) <> "" Then Kill (PicPath)
24        FileCopy .SelectedItems(1), PicPath 'Copy to new location
25        SkipCopy:
26    End With
27    CollSht.Range("B26").Value = Dir(PicPath)
28    Item_DisplayPicture
29    Item_SaveUpdate
30    NoSelection:
31 End Sub
32 Sub Item_AddTag()
33     Dim TagName As String
34     TagName = InputBox("Please enter the tag name", "Add Tag")
35     If TagName = "" Then Exit Sub
36     If CollSht.Range("B25").Value = "" Then
37         CollSht.Range("B25").Value = TagName
38     Else 'Not Empty
39         CollSht.Range("B25").Value = CollSht.Range("B25").Value & "," & TagName
40     End If
41     Item_LoadTags 'Reload tags
42 End Sub
43
44 Sub Item_ClearFilter()
45     CollSht.Range("E4,G4").ClearContents
46 End Sub
47
48 Sub Item_Delete()
49     With CollSht
50         If MsgBox("Are you sure you want to delete this Item?", vbYesNo, "Delete Item")
51             = vbNo Then Exit Sub
52         If .Range("B3").Value = Empty Then GoTo NotSaved
53         ItemRow = .Range("B3").Value 'Item Row
54         CollDB.Range("A" & ItemRow & ":V" & ItemRow).ClearContents 'Clear Clear
55         contents
56         LastRow = CollDB.Range("A99999").End(xlUp).Row
57         If LastRow < 4 Then GoTo NotSaved
58         With CollDB.Sort
59             .SortFields.Clear
60             .SortFields.Add Key:=CollDB.Range("A4"), SortOn:=xlSortOnValues, Order:=
61                 xlAscending, DataOption:=xlSortNormal 'Sort

```

1 2 3

```

1 2 3
59     .SetRange CollDB.Range("A4:V" & LastRow) 'Set Range
60     .Apply 'Apply Sort
61     End With
62     NotSaved:
63     Item_New 'Run macro to create new Item
64     Item_ListItems 'Update Item list
65     End With
66 End Sub
67
68 Sub Item_DisplayPicture()
69     With CollSht
70         .Shapes("ItemPic" ).Visible = msoFalse 'Hide Picture shape
71         If .Range("B26" ).Value = Empty Or [PictureFolder] = Empty Then Exit Sub
72         PicName = .Range("B26" ).Value
73         PicFolder = [PictureFolder]
74         PicPath = PicFolder & "\" & PicName 'Full Picture Path
75         If Dir(PicPath, vbDirectory) = "" Then Exit Sub
76         .Shapes("ItemPic" ).Fill.UserPicture PicPath
77         .Shapes("ItemPic" ).Visible = msoCTrue
78     End With
79 End Sub
80 Sub Item_FilterByTag()
81     With CollSht
82         If .Range("B24" ).Value = True Then
83             .Range("E4" ).Value = "Tags"
84             .Range("G4" ).Value = .Shapes(Application.Caller).TextFrame2.TextRange.Text
85         End If
86     End With
87 End Sub
88
89 Sub Item_ListItems()
90     For Each ItemShp In CollSht.Shapes
91         On Error Resume Next
92         If InStr(ItemShp.Name, "CollItem" ) > 0 Then ItemShp.Delete
93         If InStr(ItemShp.Name, "CollPict" ) > 0 Then ItemShp.Delete
94         On Error GoTo 0
95     Next ItemShp
96     PicFolder = [PictureFolder]
97     TopPos = CollSht.Range("D6" ).Top - 4
98     LeftPos = CollSht.Range("D6" ).Left
99     With CollDB
100        LastRow = .Range("A99999" ).End(xlUp).Row 'Last Row of data
101        If LastRow < 4 Then Exit Sub
102        .Range("A3:V" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
103            "AA2:AB3" ), CopyToRange:=.Range("AF2:AI2" ), Unique:=True
104        LastResultRow = .Range("AF99999" ).End(xlUp).Row
105        If LastResultRow < 3 Then Exit Sub
106        If LastResultRow < 4 Then GoTo SkipSort
107        With .Sort
108            .SortFields.Clear
109            .SortFields.Add Key:=CollDB.Range("AG3" ), SortOn:=xlSortOnValues, Order:=
110                xlAscending, DataOption:=xlSortNormal 'Sort
111            .SetRange CollDB.Range("AF3:AI" & LastResultRow) 'Set Range
112            .Apply 'Apply Sort
113        End With
114        SkipSort:
115        For ResultRow = 3 To LastResultRow
116            ItemID = .Range("AF" & ResultRow).Value 'Item ID
117            ItemName = .Range("AG" & ResultRow).Value 'Item Name
118            ItemDesc = .Range("AH" & ResultRow).Value 'Description
119            PicName = .Range("AI" & ResultRow).Value 'Picture

```

```

1 2 3
118 PicPath = PicFolder & "\" & PicName 'Picture Path
119 If PicFolder = "" Or PicName = "" Then PicPath = ""
120 If PicPath = "" Or Dir(PicPath, vbDirectory) = "" Then PicPath = ""
121 CollSht.Shapes("SampleItemShp").Duplicate.Name = "CollItem" & ItemID
122 With CollSht.Shapes("CollItem" & ItemID)
123     .TextFrame2.TextRange.Text = ItemName & vbCrLf & ItemDesc
124     .OnAction = "Item_Select"
125     If ResultRow Mod 2 = 0 Then 'Even
126         .Left = LeftPos + .Width + 3
127     Else 'Odd Row
128         .Left = LeftPos
129     End If
130     .Top = TopPos
131 End With
132 CollSht.Shapes("SampleItemPic").Duplicate.Name = "CollPic" & ItemID
133 With CollSht.Shapes("CollPic" & ItemID)
134     If PicPath <> "" Then .Fill.UserPicture PicPath
135     If ResultRow Mod 2 = 0 Then 'Even Row
136         .Left = LeftPos + CollSht.Shapes("CollItem" & ItemID).Width + CollSht.
137             Shapes("CollItem" & ItemID).Width - .Width
138     Else 'Odd Row
139         .Left = LeftPos + CollSht.Shapes("CollItem" & ItemID).Width - .Width
140     End If
141     .Top = TopPos + 1
142     If ResultRow Mod 2 = 0 Then TopPos = TopPos + .Height + 5
143 End With
144 Next ResultRow
145 End Sub
146 Sub Item_Load()
147     With CollSht
148         If .Range("B3").Value = Empty Then
149             MsgBox "Please select a correct Item to load"
150             Exit Sub
151         End If
152         ItemRow = .Range("B3").Value 'Item Row
153         For ItemCol = 2 To 22
154             .Range(CollDB.Cells(1, ItemCol).Value).Value = CollDB.Cells(ItemRow, ItemCol).
155                 Value 'Bring over data
156         Next ItemCol
157         Item_LoadTags
158         Attach_Refresh
159         Item_DisplayPicture
160     End With
161 Sub Item_LoadTags()
162     With CollSht
163         For Each ItemShp In .Shapes 'Remove any existing tags
164             On Error Resume Next
165             If InStr(ItemShp.Name, "ItemTag") > 0 Then ItemShp.Delete
166             If InStr(ItemShp.Name, "TagDel") > 0 Then ItemShp.Delete
167             On Error GoTo 0
168         Next ItemShp
169         If .Range("B25").Value = Empty Then Exit Sub
170         LeftPos = .Range("I26").Left 'Set Intial Left Position
171         TopPos = .Range("I26").Top + 2 'Set Intial Top Position
172         TagArr = Split(.Range("B25").Value, ",") 'Tag Array with Comma delimited
173         For TagNum = 0 To UBound(TagArr)
174             .Shapes("SampleTagShp").Duplicate.Name = "ItemTag" & TagNum + 1 'Duplicate
175                 Sample tag
176             With .Shapes("ItemTag" & TagNum + 1)

```

1 2 3 4

```

176     1 2 3 4     .TextFrame2.TextRange.Text = TagArr(TagNum)
177     .TextFrame.AutoSize = True
178     .Width = .Width + 12 'Make space for delete icon/button
179     .Height = 16
180     If LeftPos + .Width > CollSht.Range("N:N" ).Left Then
181         LeftPos = CollSht.Range("I26" ).Left 'Reset Intial Left Position
182         TopPos = TopPos + 20
183     End If
184     .Left = LeftPos
185     .Top = TopPos
186 End With
187 .Shapes("DelTagSample" ).Duplicate.Name = "TagDel" & TagNum + 1 'Duplicate
Clear Button
188 With .Shapes("TagDel" & TagNum + 1)
189     .Left = CollSht.Shapes("ItemTag" & TagNum + 1).Left + CollSht.Shapes(
"ItemTag" & TagNum + 1).Width - .Width - 2
190     .Top = TopPos + 3
191     .OnAction = "Item_RemoveTag"
192     LeftPos = LeftPos + CollSht.Shapes("ItemTag" & TagNum + 1).Width + 3
193 End With
194 Next TagNum
195 End With
196 End Sub
197
198 Sub Item_New()
199     With CollSht
200         For Each ItemShp In .Shapes 'Remove any existing tags
201             On Error Resume Next
202             If InStr(ItemShp.Name, "ItemTag" ) > 0 Then ItemShp.Delete
203             If InStr(ItemShp.Name, "TagDel" ) > 0 Then ItemShp.Delete
204             On Error GoTo 0
205         Next ItemShp
206         .Range(
"B2,B6,B25,B26,J3,M3,J5,M5,J7:M9,J11,J13,M13,J15,M15,J17,M17,J19,M19,J21,M21,J23,M
2,05:R999" ).ClearContents 'Clear Item Fields
207         .Range("B11" ).Value = False 'Set Wishlist to False
208         .Shapes("ItemPic" ).Visible = msoFalse
209         .Range("J3" ).Select
210     End With
211 End Sub
212 Sub Item_RemoveTag()
213     Dim RemTag As String
214     With CollSht
215         If MsgBox("Are you sure you want to delete this tag?" , vbYesNo, "Delete Tag" ) =
vbNo Then Exit Sub
216         If .Range("B25" ).Value = "" Then Exit Sub
217         TagNum = Replace(Application.Caller, "TagDel" , "" ) 'Extract Tag Numb
218         'TagArr = Split(.Range("B25").Value, ",") 'Tag Array with Comma delimited
219         RemTag = .Shapes("ItemTag" & TagNum).TextFrame2.TextRange.Text 'Tag Name to
Remove
220         If TagNum = 1 Then 'Remove First Tag
221             .Range("B25" ).Value = Replace(.Range("B25" ).Value, RemTag & ",", "" )
'Remove Tag and comma after
222         Else 'Remove Subsequent Tag
223             .Range("B25" ).Value = Replace(.Range("B25" ).Value, "," & RemTag, "" )
'Remove Tag and coma before
224         End If
225     End With
226     Item_LoadTags 'Reload tags
227     'Item_SaveUpdate
228 End Sub

```

```

229
230 Sub Item_SaveUpdate()
231     With CollSht
232         If .Range("J3" ).Value = "" Then
233             MsgBox "Please make sure to enter an Item Name before saving"
234             Exit Sub
235         End If
236         If .Range("B3" ).Value = Empty Then 'New Item
237             ItemRow = CollDB.Range("A99999" ).End(xlUp).Row + 1 'First Avail Row
238             .Range("B2" ).Value = .Range("B4" ).Value 'New Item. ID
239             CollDB.Range("A" & ItemRow).Value = .Range("B2" ).Value 'Set Item ID
240         Else 'Existing Item
241             ItemRow = .Range("B3" ).Value 'Existing Item Row
242         End If
243         For ItemCol = 2 To 22
244             CollDB.Cells(ItemRow, ItemCol).Value = .Range(CollDB.Cells(1, ItemCol).Value).
                Value 'Bring over data
245         Next ItemCol
246         MsgBox "Item has been saved"
247     End With
248     Item_ListItems 'Update Item list
249 End Sub
250 Sub Item_Select()
251     CollSht.Range("B2" ).Value = Replace(Application.Caller, Left(Application.Caller, 8)
        , "" )
252     Item_Load 'Run Macro To Load item
253 End Sub
254
255 Sub ItemClearPicture()
256     CollSht.Shapes("ItemPic" ).Visible = msoFalse
257 End Sub
258
259 Sub ItemPrintVisibleItems()
260     With CollSht
261         LastResultRow = CollDB.Range("AF9999" ).End(xlUp).Row
262         If LastResultRow < 3 Then
263             MsgBox "Please make sure to clear filter, or set filter, to include at least
                one item before printing"
264             Exit Sub
265         End If
266         LastRow = Int((LastResultRow / 2) * 3.8) 'Determine Last Row
267         .PageSetup.PrintArea = "$D$5:$G$" & LastRow
268         .PrintOut , , , , True, , , False
269     End With
270 End Sub

```

A

Add, 10, 11, 13, 14
 Admin, 9
 AdvancedFilter, 11, 14
 AllowMultiSelect, 9
 Application, 5, 9, 10, 13, 14, 16, 17
 Apply, 11, 14
 Attach_Add, 10
 Attach_Delete, 10
 Attach_DisplayPicture, 5, 11
 Attach_Open, 11
 Attach_Refresh, 10-12, 15
 Attach_Save, 8, 12
 AttachDB, 10-12
 AttachDesc, 10, 12
 AttachFile, 10
 AttachFrm, 10, 12
 AttachName, 10, 12
 AttachRow, 10-12
 AutoSize, 16

B

BrowseForPicFolder, 9, 13

C

Caller, 14, 16, 17
 CancelBtn_Click, 8
 Cells, 15, 17
 Clear, 11, 13, 14
 ClearContents, 11, 13, 16
 CollDB, 13-15, 17
 CollSht, 2, 10-17
 CopyToRange, 11, 14
 CountLarge, 5
 CriteriaRange, 11, 14

D

DataOption, 11, 13, 14
 Delete, 10, 14-16
 Description, 12
 Dir, 10, 11, 13-15
 Duplicate, 15, 16

E

Empty, 5, 10-15, 17
 EnableEvents, 5
 EntireRow, 10
 Explicit, 2-10, 13

F

FieldType, 2
 FileCopy, 10, 13
 FileDialog, 9, 10, 13
 FileExt, 10-12
 FileName, 10, 12
 FileType, 10, 12
 Fill, 11, 14, 15
 Filters, 10, 13
 FollowHyperlink, 11

H

Height, 15, 16

I

InputBox, 13
 InStr, 14-16
 InStrRev, 12
 Int, 17
 Intersect, 2, 5

Item_AddPicture, 13
 Item_AddTag, 13
 Item_ClearFilter, 13
 Item_Delete, 13
 Item_DisplayPicture, 13-15
 Item_FilterByTag, 14
 Item_ListItems, 5, 14, 17
 Item_Load, 15, 17
 Item_LoadTags, 13, 15, 16
 Item_New, 14, 16
 Item_RemoveTag, 16
 Item_SaveUpdate, 13, 17
 Item_Select, 17
 ItemClearPicture, 17
 ItemCol, 13, 15, 17
 ItemDesc, 13-15
 ItemID, 10, 12-15
 ItemName, 10, 12-15
 ItemPic, 13
 ItemPrintVisibleItems, 17
 ItemRow, 13, 15, 17
 ItemShp, 13-16

K

Key, 11, 13, 14
 Kill, 13

L

LastResultRow, 10, 11, 13, 14, 17
 LastRow, 10, 11, 13, 14, 17
 Left, 5, 14-17
 LeftPos, 13-16
 Len, 12

M

MsgBox, 10-13, 15-17
 msoCTrue, 5, 11, 14
 msoFalse, 5, 10, 11, 14, 16, 17
 msoFileDialogFilePicker, 10, 13
 msoFileDialogFolderPicker, 9

N

Name, 14-16
 NoSelection, 9, 10, 13
 NoSort, 11
 NotSaved, 13, 14
 NumberFormat, 2

O

OnAction, 15, 16
 Order, 11, 13, 14

P

PageSetup, 17
 PicFolder, 9-11, 13-15
 PicName, 10, 11, 13-15
 PicPath, 10, 11, 13-15
 PictureFolder, 10, 11, 13, 14
 PrintArea, 17
 PrintOut, 17

R

Range, 2, 5, 9-17
 RemTag, 16
 Replace, 16, 17
 ResultRow, 10, 13-15
 Right, 12
 Row, 2, 5, 11-14, 17
 RowNumb, 2

S

SaveBtn_Click, 8
 SelectedItems, 9, 10, 13
 SelRow, 10, 11
 SetRange, 11, 14
 Shape, 13
 Shapes, 5, 10, 11, 14-17
 Show, 9, 10, 13
 SkipCopy, 10, 13
 SkipSort, 14
 Sort, 11, 13, 14
 SortFields, 11, 13, 14
 SortOn, 11, 13, 14
 Split, 15

T

TagArr, 13, 15, 16
 TagName, 13
 TagNumb, 13, 15, 16
 Target, 2, 5
 Text, 14-16
 TextFrame, 16
 TextFrame2, 14-16
 TextRange, 14-16
 ThisWorkbook, 11
 Title, 9, 10, 13
 Top, 5, 14-16
 TopPos, 13-16

U

UBound, 15
 Unique, 11, 14
 Unload, 8, 12
 UserPicture, 11, 14, 15

V

Value, 2, 5, 9-17
 vbCrLf, 15
 vbDirectory, 11, 13-15
 vbNo, 10, 13, 16
 vbYesNo, 10, 13, 16
 Visible, 5, 10, 11, 14, 16, 17

W

Width, 15, 16
 Worksheet_Change, 2, 5
 Worksheet_SelectionChange, 5

X

xlAscending, 13, 14
 xlDescending, 11
 xlFilterCopy, 11, 14
 xlSortNormal, 11, 13, 14
 xlSortOnValues, 11, 13, 14
 xlUp, 11-14, 17

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