

# VBA SOURCE CODE BOOK

## COLLECTIBLES TRACKER

**COLLECTIBLES TRACKER**

Item Name: Paul Case    Status: On Hand

Current Value: \$279.99    Type: Display Cases

Description: Autographed Baseball with Sublimated Display Case from Fanatics Authentic.

Last Valuation: 1/21/2023    Wishlist Item

Acquired From: Ebay    Acquired On: 7/3/2021

Acquired Cost: \$223.99    Sales Price: \$484.00

Sold/Traded To: Fredders    Sale Date: 5/16/2023

Website: Fredders.com    Manufactured: Trophy Inc

First Offered: 1/15/2023    Highest Value: \$845.00

Condition: 90%    Annual Cost: \$45.00

**TAGS**

Autographed Baseball X Sublimated Display Case X  
Fanatics Authentic X Sports Memorabilia X Collector's Item X  
Signed Baseball X Display Collectibles X Sports Fanatics X  
Baseball Memorabilia X Exclusive Autograph X

**FREE TEMPLATE DOWNLOAD**



*Excel For Freelancers*

## 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](#)

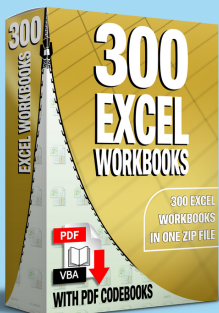
-



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](#)

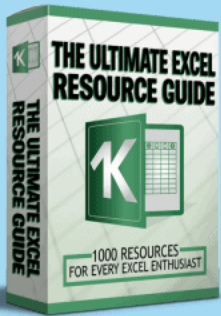
-



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](#)

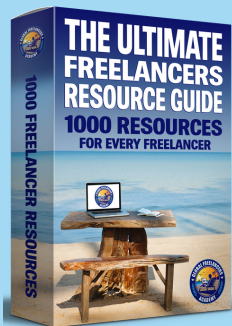
# OUR COURSES & PRODUCTS



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](#)

## Table of Contents

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
ThisWorkbook.....	6
(Declarations).....	6
Forms.....	7
AttachFrm.....	7
(Declarations).....	7
CancelBtn_Click [Sub ].....	7
SaveBtn_Click [Sub ].....	7
Modules.....	8
Admin_Macros.....	8
(Declarations).....	8
BrowseForPicFolder [Sub ].....	8
Attach_Macros.....	9
(Declarations).....	9
Attach_Add [Sub ].....	9
Attach_Delete [Sub ].....	9
Attach_DisplayPicture [Sub ].....	10
Attach_Open [Sub ].....	10
Attach_Refresh [Sub ].....	10
Attach_Save [Sub ].....	11
CollItem_Macros.....	12
(Declarations).....	12
Item_AddPicture [Sub ].....	12
Item_AddTag [Sub ].....	12
Item_ClearFilter [Sub ].....	12
Item_Delete [Sub ].....	12
Item_DisplayPicture [Sub ].....	13
Item_ListItems [Sub ].....	13
Item_Load [Sub ].....	14
Item_LoadTags [Sub ].....	14
Item_New [Sub ].....	15
Item_RemoveTag [Sub ].....	15
Item_SaveUpdate [Sub ].....	15
Item_Select [Sub ].....	16
ItemClearPicture [Sub ].....	16

```
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 Option Explicit

2



```
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 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
82 Sub Attach_Open()
83     With CollSht
84         If .Range("B6" ).Value = Empty Then
85             MsgBox "Please select an attachment to open"
86             Exit Sub
87         End If
88         SelRow = .Range("B6" ).Value
89         AttachRow = .Range("R" & SelRow).Value 'Selected Attachment Row
90         PicPath = [PictureFolder] & "\" & AttachDB.Range("C" & AttachRow).Value
91             'Attachment File Path
92         If Dir(PicPath) = "" Or PicPath = "" Then Exit Sub
93         ThisWorkbook.FollowHyperlink (PicPath) 'Open Selected File
94     End With
95 End Sub
96
97 Sub Attach_Refresh()
98     If CollSht.Range("B3" ).Value = "" Then Exit Sub 'Exit on missing id Item ID
99     CollSht.Range("B6,05:R999" ).ClearContents 'Clear Old Attachemets fields
100     CollSht.Shapes("ItemPic" ).Visible = msoFalse 'Hide Picture for now
101     With AttachDB 'With Attachments sheet
102         LastRow = .Range("A9999" ).End(xlUp).Row
103         If LastRow < 4 Then Exit Sub
104         .Range("A3:H" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
105             "K2:K3" ), CopyToRange:=.Range("N2:Q2" ), Unique:=True
106         LastResultRow = .Range("N9999" ).End(xlUp).Row 'Last Filtered Row Results
107         If LastResultRow < 3 Then Exit Sub 'If no Attach. found then attachments
108         If LastResultRow < 4 Then GoTo NoSort
109         'Sort Based on Row (Descending)
110         With .Sort
111             .SortFields.Clear
112             .SortFields.Add Key:=AttachDB.Range("Q3" ), SortOn:=xlSortOnValues, Order:=
113                 xlDescending, DataOption:=xlSortNormal 'Sort
114             .SetRange AttachDB.Range("N3:Q" & LastResultRow) 'Set Range
115             .Apply 'Apply Sort
116         End With
117         NoSort:
118         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
81 Sub Item_ListItems()
82     For Each ItemShp In CollSht.Shapes
83         On Error Resume Next
84         If InStr(ItemShp.Name, "CollItem" ) > 0 Then ItemShp.Delete
85         If InStr(ItemShp.Name, "CollPict" ) > 0 Then ItemShp.Delete
86         On Error GoTo 0
87     Next ItemShp
88     PicFolder = [PictureFolder]
89     TopPos = CollSht.Range("D6" ).Top - 4
90     LeftPos = CollSht.Range("D6" ).Left
91     With CollDB
92         LastRow = .Range("A99999" ).End(xlUp).Row 'Last Row of data
93         If LastRow < 4 Then Exit Sub
94         .Range("A3:V" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
95             "AA2:AB3" ), CopyToRange:=.Range("AF2:AI2" ), Unique:=True
96         LastResultRow = .Range("AF99999" ).End(xlUp).Row
97         If LastResultRow < 3 Then Exit Sub
98         If LastResultRow < 4 Then GoTo SkipSort
99         With .Sort
100            .SortFields.Clear
101            .SortFields.Add Key:=CollDB.Range("AG3" ), SortOn:=xlSortOnValues, Order:=
102                xlAscending, DataOption:=xlSortNormal 'Sort
103            .SetRange CollDB.Range("AF3:AI" & LastResultRow) 'Set Range
104            .Apply 'Apply Sort
105        End With
106        SkipSort:
107        For ResultRow = 3 To LastResultRow
108            ItemID = .Range("AF" & ResultRow).Value 'Item ID
109            ItemName = .Range("AG" & ResultRow).Value 'Item Name
110            ItemDesc = .Range("AH" & ResultRow).Value 'Description
111            PicName = .Range("AI" & ResultRow).Value 'Picture
112            PicPath = PicFolder & "\" & PicName 'Picture Path
113            If PicFolder = "" Or PicName = "" Then PicPath = ""
114            If PicPath = "" Or Dir(PicPath, vbDirectory) = "" Then PicPath = ""
115            CollSht.Shapes("SampleItemShp" ).Duplicate.Name = "CollItem" & ItemID
116            With CollSht.Shapes("CollItem" & ItemID)
117                .TextFrame2.TextRange.Text = ItemName & vbCrLf & ItemDesc
118                .OnAction = "Item_Select"
119            End With
120            If ResultRow Mod 2 = 0 Then 'Even

```

```

1 2 3 4 5
118     .Left = LeftPos + .Width + 3
119     }Else 'Odd Row
120     .Left = LeftPos
121     End If
122     .Top = TopPos
123     End With
124     CollSht.Shapes("SampleItemPic" ).Duplicate.Name = "CollPict" & ItemID
125     With CollSht.Shapes("CollPict" & ItemID)
126     If PicPath <> "" Then .Fill.UserPicture PicPath
127     If ResultRow Mod 2 = 0 Then 'Even Row
128     .Left = LeftPos + CollSht.Shapes("CollItem" & ItemID).Width + CollSht.
        Shapes("CollItem" & ItemID).Width - .Width
129     }Else 'Odd Row
130     .Left = LeftPos + CollSht.Shapes("CollItem" & ItemID).Width - .Width
131     End If
132     .Top = TopPos + 1
133     If ResultRow Mod 2 = 0 Then TopPos = TopPos + .Height + 5
134     End With
135     Next ResultRow
136     End With
137     End Sub
138     Sub Item_Load()
139     With CollSht
140     If .Range("B3" ).Value = Empty Then
141     MsgBox "Please select a correct Item to load"
142     Exit Sub
143     End If
144     ItemRow = .Range("B3" ).Value 'Item Row
145     For ItemCol = 2 To 22
146     .Range(CollDB.Cells(1, ItemCol).Value).Value = CollDB.Cells(ItemRow, ItemCol).
        Value 'Bring over data
147     Next ItemCol
148     Item_LoadTags
149     Attach_Refresh
150     Item_DisplayPicture
151     End With
152     End Sub
153     Sub Item_LoadTags()
154     With CollSht
155     For Each ItemShp In .Shapes 'Remove any existing tags
156     On Error Resume Next
157     If InStr(ItemShp.Name, "ItemTag" ) > 0 Then ItemShp.Delete
158     If InStr(ItemShp.Name, "TagDel" ) > 0 Then ItemShp.Delete
159     On Error GoTo 0
160     Next ItemShp
161     If .Range("B25" ).Value = Empty Then Exit Sub
162     LeftPos = .Range("I26" ).Left 'Set Intial Left Position
163     TopPos = .Range("I26" ).Top + 2 'Set Intial Top Position
164     TagArr = Split(.Range("B25" ).Value, "," ) 'Tag Array with Comma delimited
165     For TagNumb = 0 To UBound(TagArr)
166     .Shapes("SampleTagShp" ).Duplicate.Name = "ItemTag" & TagNumb + 1 'Duplicate
        Sample tag
167     With .Shapes("ItemTag" & TagNumb + 1)
168     .TextFrame2.TextRange.Text = TagArr(TagNumb)
169     .TextFrame.AutoSize = True
170     .Width = .Width + 12 'Make space for delete icon/button
171     .Height = 16
172     If LeftPos + .Width > CollSht.Range("N:N" ).Left Then
173     LeftPos = CollSht.Range("I26" ).Left 'Reset Intial Left Position
174     TopPos = TopPos + 20
175     End If

```

```

176     1 2 3 4     .Left = LeftPos
177     .Top = TopPos
178     End With
179     .Shapes("DelTagSample" ).Duplicate.Name = "TagDel" & TagNum + 1 'Duplicate
Clear Button
180     With .Shapes("TagDel" & TagNum + 1)
181     .Left = CollSht.Shapes("ItemTag" & TagNum + 1).Left + CollSht.Shapes(
"ItemTag" & TagNum + 1).Width - .Width - 2
182     .Top = TopPos + 3
183     .OnAction = "Item_RemoveTag"
184     LeftPos = LeftPos + CollSht.Shapes("ItemTag" & TagNum + 1).Width + 3
185     End With
186     Next TagNum
187     End With
188     End Sub

189
190     Sub Item_New()
191     With CollSht
192     For Each ItemShp In .Shapes 'Remove any existing tags
193     On Error Resume Next
194     If InStr(ItemShp.Name, "ItemTag" ) > 0 Then ItemShp.Delete
195     If InStr(ItemShp.Name, "TagDel" ) > 0 Then ItemShp.Delete
196     On Error GoTo 0
197     Next ItemShp
198     .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
199     .Range("B11" ).Value = False 'Set Wishlist to False
200     .Shapes("ItemPic" ).Visible = msoFalse
201     .Range("J3" ).Select
202     End With
203     End Sub

204     Sub Item_RemoveTag()
205     Dim RemTag As String
206     With CollSht
207     If MsgBox("Are you sure you want to delete this tag?" , vbYesNo, "Delete Tag" ) =
vbNo Then Exit Sub
208     If .Range("B25" ).Value = "" Then Exit Sub
209     TagNum = Replace(Application.Caller, "TagDel" , "" ) 'Extract Tag Num
210     'TagArr = Split(.Range("B25").Value, ",") 'Tag Array with Comma delimited
211     RemTag = .Shapes("ItemTag" & TagNum).TextFrame2.TextRange.Text 'Tag Name to
Remove
212     If TagNum = 1 Then 'Remove First Tag
213     .Range("B25" ).Value = Replace(.Range("B25" ).Value, RemTag & ",", "" )
'Remove Tag and comma after
214     Else 'Remove Subsequent Tag
215     .Range("B25" ).Value = Replace(.Range("B25" ).Value, "," & RemTag, "" )
'Remove Tag and coma before
216     End If
217     End With
218     Item_LoadTags 'Reload tags
219     'Item_SaveUpdate
220     End Sub

221
222     Sub Item_SaveUpdate()
223     With CollSht
224     If .Range("J3" ).Value = "" Then
225     MsgBox "Please make sure to enter an Item Name before saving"
226     Exit Sub
227     End If
228     If .Range("B3" ).Value = Empty Then 'New Item

```

```
1 2 3
229 ItemRow = CollDB.Range("A99999" ).End(xlUp).Row + 1 'First Avail Row
230 .Range("B2" ).Value = .Range("B4" ).Value 'New Item. ID
231 CollDB.Range("A" & ItemRow).Value = .Range("B2" ).Value 'Set Item ID
232 Else 'Existing Item
233 ItemRow = .Range("B3" ).Value 'Existing Item Row
234 End If
235 For ItemCol = 2 To 22
236 CollDB.Cells(ItemRow, ItemCol).Value = .Range(CollDB.Cells(1, ItemCol).Value).
Value 'Bring over data
237 Next ItemCol
238 MsgBox "Item has been saved"
239 End With
240 Item_ListItems 'Update Item list
241 End Sub
242 Sub Item_Select()
243 CollSht.Range("B2" ).Value = Replace(Application.Caller, Left(Application.Caller, 8)
, "" )
244 Item_Load 'Run Macro To Load item
245 End Sub
246
247 Sub ItemClearPicture()
248 CollSht.Shapes("ItemPic" ).Visible = msoFalse
249 End Sub
```

**A**

Add, 9, 10, 12, 13  
 Admin, 8  
 AdvancedFilter, 10, 13  
 AllowMultiSelect, 8  
 Application, 5, 8, 9, 12, 15, 16  
 Apply, 10, 13  
 Attach\_Add, 9  
 Attach\_Delete, 9  
 Attach\_DisplayPicture, 5, 10  
 Attach\_Open, 10  
 Attach\_Refresh, 9-11, 14  
 Attach\_Save, 7, 11  
 AttachDB, 9-11  
 AttachDesc, 9, 11  
 AttachFile, 9  
 AttachFrm, 9, 11  
 AttachName, 9, 11  
 AttachRow, 9-11  
 AutoSize, 14

**B**

BrowseForPicFolder, 8, 12

**C**

Caller, 15, 16  
 CancelBtn\_Click, 7  
 Cells, 14, 16  
 Clear, 10, 12, 13  
 ClearContents, 10, 12, 15  
 CollDB, 12-14, 16  
 CollSht, 2, 9-16  
 CopyToRange, 10, 13  
 CountLarge, 5  
 CriteriaRange, 10, 13

**D**

DataOption, 10, 12, 13  
 Delete, 9, 13-15  
 Description, 11  
 Dir, 9, 10, 12, 13  
 Duplicate, 13-15

**E**

Empty, 5, 9-15  
 EnableEvents, 5  
 EntireRow, 9  
 Explicit, 2-9, 12

**F**

FieldType, 2  
 FileCopy, 9, 12  
 FileDialog, 8, 9, 12  
 FileExt, 9-11  
 FileName, 9, 11  
 FileType, 9, 11  
 Fill, 10, 13, 14  
 Filters, 9, 12  
 FollowHyperlink, 10

**H**

Height, 14

**I**

InputBox, 12  
 InStr, 13-15  
 InStrRev, 11  
 Intersect, 2, 5  
 Item\_AddPicture, 12

Item\_AddTag, 12  
 Item\_ClearFilter, 12  
 Item\_Delete, 12  
 Item\_DisplayPicture, 12-14  
 Item\_ListItems, 5, 13, 16  
 Item\_Load, 14, 16  
 Item\_LoadTags, 12, 14, 15  
 Item\_New, 13, 15  
 Item\_RemoveTag, 15  
 Item\_SaveUpdate, 12, 15  
 Item\_Select, 16  
 ItemClearPicture, 16  
 ItemCol, 12, 14, 16  
 ItemDesc, 12, 13  
 ItemID, 9, 11-14  
 ItemName, 9, 11-13  
 ItemPic, 12  
 ItemRow, 12, 14, 16  
 ItemShp, 12-15

**K**

Key, 10, 12, 13  
 Kill, 12

**L**

LastResultRow, 9, 10, 12, 13  
 LastRow, 9, 10, 12, 13  
 Left, 5, 13-16  
 LeftPos, 12-15  
 Len, 11

**M**

MsgBox, 9-12, 14-16  
 msoCTrue, 5, 10, 13  
 msoFalse, 5, 9, 10, 13, 15, 16  
 msoFileDialogFilePicker, 9, 12  
 msoFileDialogFolderPicker, 8

**N**

Name, 13-15  
 NoSelection, 8, 9, 12  
 NoSort, 10  
 NotSaved, 12, 13  
 NumberFormat, 2

**O**

OnAction, 13, 15  
 Order, 10, 12, 13

**P**

PicFolder, 8-10, 12, 13  
 PicName, 9, 10, 12, 13  
 PicPath, 9, 10, 12-14  
 PictureFolder, 9, 10, 12, 13

**R**

Range, 2, 5, 8-16  
 RemTag, 15  
 Replace, 15, 16  
 ResultRow, 9, 12-14  
 Right, 11  
 Row, 2, 5, 10-13, 16  
 RowNumb, 2

**S**

SaveBtn\_Click, 7  
 SelectedItems, 8, 9, 12  
 SelRow, 9, 10  
 SetRange, 10, 13  
 Shape, 12

Shapes, 5, 9, 10, 13-16  
 Show, 8, 9, 12  
 SkipCopy, 9, 12  
 SkipSort, 13  
 Sort, 10, 12, 13  
 SortFields, 10, 12, 13  
 SortOn, 10, 12, 13  
 Split, 14

**T**

TagArr, 12, 14  
 TagName, 12  
 TagNumb, 12, 14, 15  
 Target, 2, 5  
 Text, 13-15  
 TextFrame, 14  
 TextFrame2, 13-15  
 TextRange, 13-15  
 ThisWorkbook, 10  
 Title, 8, 9, 12  
 Top, 5, 13-15  
 TopPos, 12-15

**U**

UBound, 14  
 Unique, 10, 13  
 Unload, 7, 11  
 UserPicture, 10, 13, 14

**V**

Value, 2, 5, 8-16  
 vbCrLf, 13  
 vbDirectory, 10, 12, 13  
 vbNo, 9, 12, 15  
 vbYesNo, 9, 12, 15  
 Visible, 5, 9, 10, 13, 15, 16

**W**

Width, 14, 15  
 Worksheet\_Change, 2, 5  
 Worksheet\_SelectionChange, 5

**X**

xlAscending, 12, 13  
 xlDescending, 10  
 xlFilterCopy, 10, 13  
 xlSortNormal, 10, 12, 13  
 xlSortOnValues, 10, 12, 13  
 xlUp, 10-13, 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*