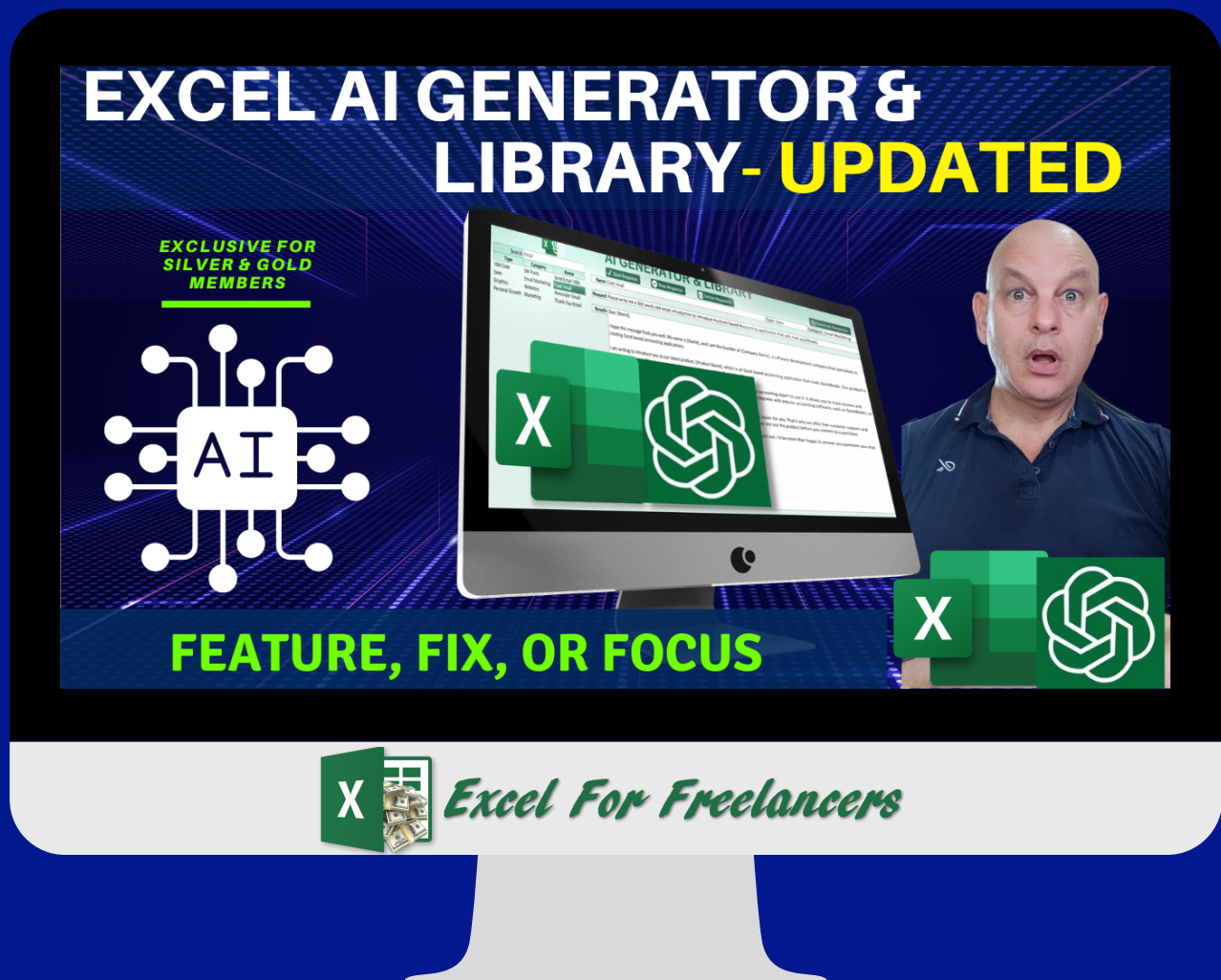


VBA SOURCE CODE BOOK



Create The **ULTIMATE** AI
Generator & Library With
CHAT GPT & 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 442,000 YouTube subscribers, 33,000,000 video views, 250+ comprehensive training videos, and a thriving 60,000 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



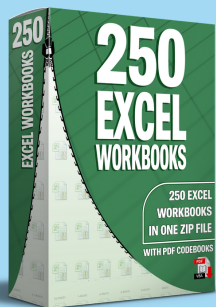
This comprehensive program will take you through a 12-phase process that will turn your enthusiasm for Excel into passive income.

[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 250 of my BEST Applications now with PDF VBA Codebooks packed into a SINGLE ZIP File which also includes the "250 Workbook Library".

[Click here to learn more](#)



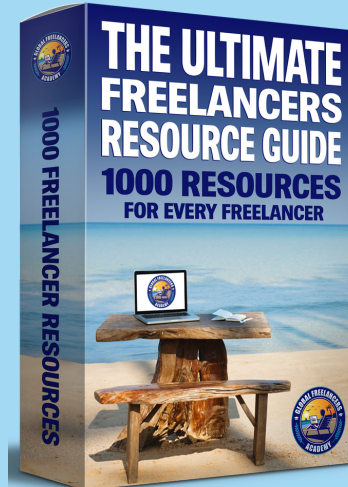
With 1000 live links, continuously updating content, sort-able and filterable items, you will always have exactly what you need, when you need it.

[Click here to learn more](#)



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 along with a ton of resources for anyone to reach their goals through freelancing.

[Click here to learn more](#)



I've put together all the 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](#)

Table of Contents

Projects.....	2
VBAProject.....	2
Documents.....	2
AIGen.....	2
(Declarations).....	2
Worksheet_Change [Sub].....	2
Worksheet_SelectionChange [Sub].....	2
RespList.....	3
(Declarations).....	3
ThisWorkbook.....	4
(Declarations).....	4
Modules.....	5
AICode_Macros.....	5
(Declarations).....	5
AIResp_Delete [Sub].....	5
AIResp_GetResponse [Sub].....	5
AIResp_GetUniqueCategories [Sub].....	6
AIResp_GetUniqueNames [Sub].....	7
AIResp_GetUniqueTypes [Sub].....	7
AIResp_Load [Sub].....	8
AIResp_New [Sub].....	8
AIResp_RunSearch [Sub].....	8
AIResp_Save [Sub].....	8
AIResp_ShowPicture [Sub].....	9
BrowseForPicFolder [Sub].....	9
CopyToClipboard [Sub].....	9

```
1 Option Explicit
2
3 Private Sub Worksheet_Change(ByVal Target As Range)
4     If Not Intersect(Target, Range("E2" )) Is Nothing And Range("E2" ).Value <> Empty
5         Then AIResp_RunSearch
6     End Sub
7
8 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
9     If Target.CountLarge > 1 Then Exit Sub
10    If Not Intersect(Target, Range("F4:F99" )) Is Nothing And Range("F" & Target.Row).
11    Value <> Empty Then
12        Range("B7" ).Value = Target.Row
13        Range("B2" ).Value = Range("G" & Target.Row).Value 'Set Resp. ID
14        AIResp_Load
15    End If
16
17    'On type Selection, load associated categories
18    If Not Intersect(Target, Range("D4:D999" )) Is Nothing And Range("D" & Target.Row).
19    Value <> Empty Then
20        Range("B5" ).Value = Target.Row
21        AIResp_GetUniqueCategories 'Get Unique Cagegories
22    End If
23
24    'On Selection of Category, load unique names
25    If Not Intersect(Target, Range("E4:E999" )) Is Nothing And Range("E" & Target.Row).
26    Value <> Empty Then
27        Range("B6" ).Value = Target.Row
28        AIResp_GetUniqueNames 'Get Unique Names
29    End If
30 End Sub
```

1 Option Explicit

2

1 Option Explicit

2


```

1 Option Explicit
2 #If VBA7 And Win64 Then
3 Private Declare PtrSafe Function URLDownloadToFile Lib "urlmon" _
4 Alias "URLDownloadToFileA" (ByVal pCaller As Long, _
5 ByVal szURL As String, ByVal szFileName As String, _
6 ByVal dwReserved As LongPtr, ByVal lpfnCB As Long) As Long
7 #Else
8 Private Declare Function URLDownloadToFile Lib "urlmon" _
9 Alias "URLDownloadToFileA" ( _
10 ByVal pCaller As Long, _
11 ByVal szURL As String, _
12 ByVal szFileName As String, _
13 ByVal dwReserved As Long, _
14 ByVal lpfnCB As Long _
15 ) As Long
16 #End If
17
18 Dim LastRow As Long, RespRow As Long, RespCol As Long, LastResultRow As Long, TypeRow
19 As Long, CatRow As Long
20
21 Sub AIResp_Delete()
22     If MsgBox("Are you sure you want to delete this response?" , vbYesNo, "Delete
23     Response" ) = vbNo Then Exit Sub
24     With AIGen
25         If .Range("B3" ).Value = Empty Then GoTo NotSaved
26         RespRow = .Range("B3" ).Value 'Response Row
27         RespList.Range(RespRow & ":" & RespRow).EntireRow.Delete
28         On Error Resume Next
29         .Shapes("AIGenPic" ).Delete 'Remove Picture if it exist
30         On Error GoTo 0
31         NotSaved:
32         AIResp_New
33         AIResp_GetUniqueTypes 'Run Macro To Get Unique types
34         AIResp_GetUniqueCategories 'Run Macro To Get Unique Categories
35         AIResp_GetUniqueNames 'Run Macro To Get Unique Name
36     End With
37 End Sub
38
39 ' Revised procedure Sammy
40 Sub AIResp_GetResponse()
41     Dim oXMLHTTP As Object
42     Dim sResponse As String, sBody As String, MyRequest As String, Resp As String,
43     ReqType As String
44     Dim SecKey As String, PicFolder As String, FileName As String, PicPath As String,
45     PicURL As String
46     With AIGen
47         If .Range("M5" ).Value = Empty Then
48             MsgBox "Please select a Request Type"
49             Exit Sub
50         End If
51
52         If .Range("P3" ).Value = Empty Then
53             MsgBox "Please make sure to add in your secret key"
54         End If
55
56         SecKey = .Range("P3" ).Value 'Secret Key
57         ReqType = .Range("M5" ).Value 'Request Type
58
59         On Error Resume Next
60         .Shapes("AIGenPic" ).Delete 'Remove Picture if it exist
61         On Error GoTo 0

```

```

58
59 AIGen.Range("I7:M33").ClearContents
60 MyRequest = .Range("I5").Value 'Requested Text
61 Set oXMLHTTP = CreateObject("MSXML2.ServerXMLHTTP")
62
63 If ReqType = "Text" Then 'Text Request
64     oXMLHTTP.Open "POST", "https://api.openai.com/v1/completions", False 'Set
        Open Posts For Text
65     sBody = "{" & "model" & ": " & "text-davinci-003" & ", " & "prompt" & ": " & "" & MyRequest
        & "" & ", " & "temperature" & ": 0, " & "max_tokens" & ": 3500}" 'Request Body For
        Text Response
66 Else 'Picture Request
67     PicFolder = .Range("P4").Value 'Picture Folder
68     If Dir(PicFolder, vbDirectory) = "" Or PicFolder = Empty Then
69         MsgBox "Please select a folder to add pictures before requesting pictures"
70
71         BrowseForPicFolder 'Browse For Picture Folder
72         PicFolder = .Range("P4").Value 'Reset Picture variable
73         If Dir(PicFolder, vbDirectory) = "" Or PicFolder = Empty Then Exit Sub
74         'Exit
75     End If
76     oXMLHTTP.Open "POST", "https://api.openai.com/v1/images/generations", False
        'Set Open Post For Images
77     sBody = "{" & "model" & ": " & "image-alpha-001" & ", " & "prompt" & ": " & "" & MyRequest
        & "" & "}"
78     End If
79     oXMLHTTP.setRequestHeader "Content-Type", "application/json" 'Set Header Type
80     oXMLHTTP.setRequestHeader "Authorization", "Bearer " & SecKey 'Set Header
        Authorization
81     oXMLHTTP.Send sBody
82
83     Resp = oXMLHTTP.responseText 'Extract Response
84     If ReqType = "Text" Then 'Text Response
85         .Range("I7").Value = Mid(Resp, InStr(Resp, "\n\n") + 4, InStr(Resp, Chr(34)
            & ", " & Chr(34) & "index" ) - InStr(Resp, "\n\n") - 4)
86     Else 'Picture Request
87         PicURL = Mid(Resp, InStr(Resp, "url" ) + 7, InStr(Resp, "}" ) - InStr(Resp,
            "url" ) - 13)
88         FileName = Format(Now(), "YYYY-MM-DD-HH-SS" ) & ".png" 'Set Unique File Name
89
90         PicPath = PicFolder & "\" & FileName
91         URLDownloadToFile 0, PicURL, PicPath, 0, 0
92         .Range("B9").Value = PicPath
93         AIResp_ShowPicture 'Run Macro To Display Picture
94     End If
95 End With
96 End Sub
97
98 Sub AIResp_GetUniqueCategories()
99     AIGen.Range("B6:B7,E4:E999").ClearContents 'Clear existing Categories
100     With Resplist
101         LastRow = .Range("A99999").End(xlUp).Row 'Last Row
102         If LastRow < 4 Then Exit Sub
103         If AIGen.Range("B5").Value = Empty Then Exit Sub
104         TypeRow = AIGen.Range("B5").Value 'Selected Type Row
105         .Range("M3").Value = AIGen.Range("D" & TypeRow).Value ' Set Type Criteria
106         .Range("C3:H" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
            "M2:M3" ), CopyToRange:=.Range("P2" ), Unique:=True
107         LastResultRow = .Range("P99999").End(xlUp).Row
108         If LastResultRow < 3 Then Exit Sub

```

```

1 2
107     AIGen.Range("E4:E" & LastResultRow + 1).Value = .Range("P3:P" & LastResultRow).
        Value 'Bring Over Category Results
108     End With
109
110     'Select Displayed Category if it exists
111     If AIGen.Range("M3" ).Value <> Empty Then
112         On Error Resume Next
113         AIGen.Range("B6" ).Value = AIGen.Range("E4:E9999" ).Find(AIGen.Range("M3" ).Value
        , , xlValues, xlWhole).Row
114         On Error GoTo 0
115     End If
116 End Sub
117
118 Sub AIResp_GetUniqueNames()
119     AIGen.Range("B7,F4:G999" ).ClearContents 'Clear existing Categories
120     With Resplist
121         LastRow = .Range("A99999" ).End(xlUp).Row 'Last Row
122         If LastRow < 4 Then Exit Sub
123         If AIGen.Range("B6" ).Value = Empty Then Exit Sub
124         CatRow = AIGen.Range("B6" ).Value 'Selected Category Row
125         .Range("N3" ).Value = AIGen.Range("E" & CatRow).Value ' Set Category Criteria
126         .Range("A3:D" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
        "M2:N3" ), CopyToRange:=.Range("R2:S2" ), Unique:=True
127         LastResultRow = .Range("R99999" ).End(xlUp).Row
128         If LastResultRow < 3 Then Exit Sub
129         AIGen.Range("F4:G" & LastResultRow + 1).Value = .Range("R3:S" & LastResultRow).
        Value 'Bring Over Category Results
130     End With
131
132     'Select Displayed Name if it exists
133     If AIGen.Range("I3" ).Value <> Empty Then
134         On Error Resume Next
135         AIGen.Range("B7" ).Value = AIGen.Range("F4:F9999" ).Find(AIGen.Range("I3" ).Value
        , , xlValues, xlWhole).Row 'Locate Name Row
136         On Error GoTo 0
137     End If
138 End Sub
139
140 Sub AIResp_GetUniqueTypes()
141     AIGen.Range("B5:B7,D4:D999" ).ClearContents 'Clear out selected type, category &
        Name & Types
142     With Resplist
143         LastRow = .Range("A99999" ).End(xlUp).Row 'Last Row
144         If LastRow < 4 Then Exit Sub
145         On Error Resume Next
146         .Names("Criteria" ).Delete
147         On Error GoTo 0
148         .Range("C3:C" & LastRow).AdvancedFilter xlFilterCopy, , CopyToRange:=.Range(
        "K2" ), Unique:=True
149         LastResultRow = .Range("K99999" ).End(xlUp).Row
150         If LastResultRow < 3 Then Exit Sub
151         AIGen.Range("D4:D" & LastResultRow + 1).Value = .Range("K3:K" & LastResultRow).
        Value 'Bring Over Type Results
152     End With
153
154     'Select Displayed type if it exists
155     If AIGen.Range("K3" ).Value <> Empty Then
156         On Error Resume Next
157         AIGen.Range("B5" ).Value = AIGen.Range("D4:D9999" ).Find(AIGen.Range("K3" ).Value
        , , xlValues, xlWhole).Row
158         On Error GoTo 0

```

1 2

```

159     End If
160 End Sub
161
162 Sub AIResp_Load()
163     With AIGen
164         If .Range("B3" ).Value = Empty Then
165             MsgBox "Please select a correct response"
166             Exit Sub
167         End If
168         .Range("B9,E2:F2,I3,K3,M3,I5:K5,I7:M33" ).ClearContents
169         On Error Resume Next
170         .Shapes("AIGenPic" ).Delete 'Remove Picture if it exist
171         On Error GoTo 0
172         RespRow = .Range("B3" ).Value 'Response Row
173         For RespCol = 2 To 8
174             .Range(RespList.Cells(1, RespCol).Value).Value = RespList.Cells(RespRow,
175                 RespCol).Value 'Load Data From List
176         Next RespCol
177         If .Range("M5" ).Value = "Picture" And .Range("B9" ).Value <> Empty Then
178             AIResp_ShowPicture
179         End With
180     End Sub
181
182 Sub AIResp_New()
183     AIGen.Range("B2,B7,B9,E2:F2,I3,K3,M3,I5:K5,I7:M33" ).ClearContents
184     On Error Resume Next
185     AIGen.Shapes("AIGenPic" ).Delete 'Remove Picture if it exist
186     On Error GoTo 0
187 End Sub
188
189 Sub AIResp_RunSearch()
190     AIGen.Range("B5:B7,F4:G999" ).ClearContents
191     With RespList
192         LastRow = .Range("A99999" ).End(xlUp).Row 'Last Row
193         If LastRow < 4 Then Exit Sub
194         .Range("A3:D" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
195             "V2:V3" ), CopyToRange:=.Range("R2:S2" ), Unique:=True
196         LastResultRow = .Range("R99999" ).End(xlUp).Row
197         If LastResultRow < 3 Then Exit Sub
198         AIGen.Range("F4:G" & LastResultRow + 1).Value = .Range("R3:S" & LastResultRow).
199             Value 'Bring Over Category Results
200     End With
201 End Sub
202
203 ' Revised procedure Sammy
204
205 Sub AIResp_Save()
206     With AIGen
207         If .Range("B8" ).Value < 5 Then
208             MsgBox "Please make sure to add in a Name,Type, Category, Return Type &
209                 Request before saving"
210             Exit Sub
211         End If
212         If .Range("B3" ).Value = Empty Then
213             RespRow = RespList.Range("A99999" ).End(xlUp).Row + 1 'First Avail Row
214             .Range("B2" ).Value = .Range("B4" ).Value 'Next Avail. ID
215             RespList.Range("A" & RespRow).Value = .Range("B2" ).Value 'Resp. ID
216         Else 'Existing
217             RespRow = .Range("B3" ).Value 'Existing Row
218         End If
219         For RespCol = 2 To 8

```

```
1 2 3
215   RespList.Cells(RespRow, RespCol).Value = .Range(RespList.Cells(1, RespCol).
      Value).Value 'Save Data To List
216   Next RespCol
217   MsgBox "Response Saved"
218   RespList.Rows.RowHeight = 15
219   AIResp_GetUniqueTypes 'Run Macro To Get Unique types
220   AIResp_GetUniqueCategories 'Run Macro To Get Unique Categories
221   AIResp_GetUniqueNames 'Run Macro To Get Unique Name
222   End With
223 End Sub
224
225 Sub AIResp_ShowPicture()
226   Dim PicPath As String
227   With AIGen
228     PicPath = .Range("B9" ).Value
229     If Dir(PicPath, vbDirectory) = "" Then Exit Sub
230     .Pictures.Insert(PicPath).Name = "AIGenPic"
231     With AIGen.Shapes("AIGenPic" )
232       .LockAspectRatio = msoCTrue
233       If .Width > .Height Then .Width = 550 Else .Height = 350
234       .Left = AIGen.Range("I7" ).Left + ((AIGen.Range("I:M" ).Width - .Width) / 2)
235       .Top = AIGen.Range("I7" ).Top + ((AIGen.Range("7:33" ).Height - .Height) / 2)
236     End With
237   End With
238 End Sub
239
240
241 Sub BrowseForPicFolder()
242   Dim PicFolder As FileDialog
243   Set PicFolder = Application.FileDialog(msoFileDialogFolderPicker)
244   With PicFolder
245     .Title = "Browse for item picture folder"
246     .AllowMultiSelect = False
247     If .Show <> -1 Then GoTo NoSelection
248     AIGen.Range("P4" ) = .SelectedItems(1)
249   End With
250   NoSelection:
251 End Sub
252
253 Sub CopyToClipboard()
254   AIGen.Range("I7:M33" ).Copy
255 End Sub
256
257
```

—
_, 5

A

AdvancedFilter, 6-8
 AIGen, 5-9
 AIResp_Delete, 5
 AIResp_GetResponse, 5
 AIResp_GetUniqueCategories, 2, 5, 6, 9
 AIResp_GetUniqueNames, 2, 5, 7, 9
 AIResp_GetUniqueTypes, 5, 7, 9
 AIResp_Load, 2, 8
 AIResp_New, 5, 8
 AIResp_RunSearch, 2, 8
 AIResp_Save, 8
 AIResp_ShowPicture, 6, 8, 9
 AllowMultiSelect, 9
 Application, 9

B

BrowseForPicFolder, 6, 9

C

CatRow, 5, 7
 Cells, 8, 9
 Chr, 6
 ClearContents, 6-8
 Copy, 9
 CopyToClipboard, 9
 CopyToRange, 6-8
 CountLarge, 2
 CreateObject, 6
 CriteriaRange, 6-8

D

Delete, 5, 7, 8
 Dir, 6, 9
 dwReserved, 5

E

Empty, 2, 5-8
 EntireRow, 5
 Explicit, 2-5

F

FileDialog, 9
 FileName, 5, 6
 Find, 7
 Format, 6

H

Height, 9

I

Insert, 9
 InStr, 6
 Intersect, 2

L

LastResultRow, 5-8
 LastRow, 5-8
 Left, 9
 LockAspectRatio, 9
 LongPtr, 5
 lpfnCB, 5

M

Mid, 6
 MsgBox, 5, 6, 8, 9

msoCTrue, 9
 msoFileDialogFolderPicker, 9
 MyRequest, 5, 6

N

Name, 9
 Names, 7
 NoSelection, 9
 NotSaved, 5
 Now, 6

O

Open, 6
 oXMLHTTP, 5, 6

P

pCaller, 5
 PicFolder, 5, 6, 9
 PicPath, 5, 6, 9
 Pictures, 9
 PicURL, 5, 6
 PtrSafe, 5

R

Range, 2, 5-9
 ReqType, 5, 6
 Resp, 5, 6
 RespCol, 5, 8, 9
 Resplist, 5-9
 responseText, 6
 RespRow, 5, 8, 9
 Row, 2, 6-8
 RowHeight, 9
 Rows, 9

S

sBody, 5, 6
 SecKey, 5, 6
 SelectedItems, 9
 Send, 6
 SetRequestHeader, 6
 Shapes, 5, 8, 9
 Show, 9
 sResponse, 5
 szFileName, 5
 szURL, 5

T

Target, 2
 Title, 9
 Top, 9
 TypeRow, 5, 6

U

Unique, 6-8
 URLDownloadToFile, 5, 6

V

Value, 2, 5-9
 VBA7, 5
 vbDirectory, 6, 9
 vbNo, 5
 vbYesNo, 5

W

Width, 9
 Win64, 5
 Worksheet_Change, 2
 Worksheet_SelectionChange, 2

X

xlFilterCopy, 6-8
 xlUp, 6-8
 xlValues, 7
 xlWhole, 7

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