

VBA SOURCE CODE BOOK



The 4 Best AI Tools You Have
EVER Seen In Excel



[DOWNLOAD
APPLICATION](#)



[VIEW
TRAINING](#)

by: *Randy Austin*

ABOUT THE AUTHOR

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

With over 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



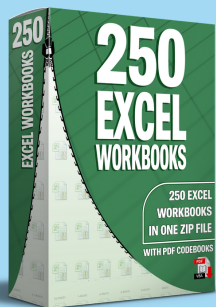
This comprehensive program will take you through a 12-phase process that will turn your enthusiasm for Excel into passive income.

[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](#)



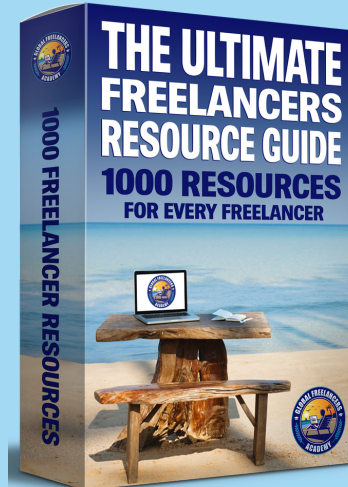
With 1000 live links, continuously updating content, sort-able and filterable items, you will always have exactly what you need, when you need it.

[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](#)

Projects.....	2
VBAProject.....	2
Documents.....	2
Admin.....	2
(Declarations).....	2
Sheet1.....	3
(Declarations).....	3
Sheet2.....	4
(Declarations).....	4
Sheet3.....	5
(Declarations).....	5
ThisWorkbook.....	6
(Declarations).....	6
Workbook_Deactivate [Sub].....	6
Workbook_SheetBeforeRightClick [Sub].....	6
Forms.....	7
FixMyCodeForm.....	7
(Declarations).....	7
CloseBtn_Click [Sub].....	7
FixMyCodeBtn_Click [Sub].....	7
FixMyFormulaForm.....	8
(Declarations).....	8
CloseBtn_Click [Sub].....	8
Modules.....	9
ChatGPT_Function.....	9
(Declarations).....	9
ChatGPT [Function].....	9
DataInTable_Macro.....	10
(Declarations).....	10
ReturnDataInTable [Sub].....	10
FixMyCode_Macro.....	12
(Declarations).....	12
FixMyCode [Sub].....	12
FixMyCode_GetResponse [Sub].....	12
PrintSheet [Sub].....	13
Send_Email [Sub].....	13
FixMyFormula_Macro.....	14
(Declarations).....	14
FixMyFormula [Sub].....	14

1
2

Option Explicit

1
2

Option Explicit

1
2

Option Explicit

1
2

Option Explicit

```
1 Option Explicit
2
3 Private Sub Workbook_Deactivate()
4     On Error Resume Next
5     With Application
6         .CommandBars("Cell" ).Controls("ChatGPT Table Results" ).Delete
7         .CommandBars("Cell" ).Controls("Fix My Formula" ).Delete
8         .CommandBars("Cell" ).Controls("Fix My Code" ).Delete
9     End With
10    On Error GoTo 0
11 End Sub
12
13 Private Sub Workbook_SheetBeforeRightClick(ByVal Sh As Object, ByVal Target As Range,
14 Cancel As Boolean)
15     Dim cmdBtn As CommandBarButton, cmdBtn2 As CommandBarButton, cmdBtn3 As
16     CommandBarButton
17     On Error Resume Next
18     With Application
19         .CommandBars("Cell" ).Controls("ChatGPT Table Results" ).Delete
20         .CommandBars("Cell" ).Controls("Fix My Formula" ).Delete
21         .CommandBars("Cell" ).Controls("Fix My Code" ).Delete
22         Set cmdBtn = .CommandBars("Cell" ).Controls.Add(Temporary:=True, Before:=1, Type:
23         =msoControlButton)
24         Set cmdBtn2 = .CommandBars("Cell" ).Controls.Add(Temporary:=True, Before:=1, Type
25         :=msoControlButton)
26         Set cmdBtn3 = .CommandBars("Cell" ).Controls.Add(Temporary:=True, Before:=1, Type
27         :=msoControlButton)
28     End With
29     With cmdBtn
30         .Caption = "ChatGPT Table Results"
31         .FaceId = 6180
32         .OnAction = "ReturnDataInTable"
33     End With
34     With cmdBtn2
35         .Caption = "Fix My Formula"
36         .FaceId = 11938
37         .OnAction = "FixMyFormula"
38     End With
39     With cmdBtn3
40         .Caption = "Fix My Code"
41         .FaceId = 7738
42         .OnAction = "FixMyCode"
43     End With
44     On Error GoTo 0
45 End Sub
```

```
1 Option Explicit
2
3 Private Sub CloseBtn_Click()
4     FixMyCodeForm.Hide
5 End Sub
6
7 Private Sub FixMyCodeBtn_Click()
8     FixMyCode_GetResponse 'Run Macro To Generate response
9 End Sub
10
11
```

```
1 Option Explicit
2
3 Private Sub CloseBtn_Click()
4     Me.Hide
5 End Sub
```

Option Explicit

```

Function ChatGPT(Prompt As String) As String
    Dim oXMLHTTP As Object
    Dim AIBody As String, Response As String, SecretKey As String, ParsedAnswer As String, Temp As String, Model As String
    SecretKey = Admin.Range("D5").Value 'Secret Key
    Model = Admin.Range("D6").Value 'Model
    Temp = Admin.Range("D7").Value 'Temperature
    If Model = "" Then Model = "text-davinci-003" 'Set Default Model
    If Temp = "" Then Temp = "0.4" 'Set Default Temp
    Prompt = Replace(Replace(Replace(Prompt, Chr(34), "\" & Chr(34)), vbNewLine, "\n"), Chr(10), "\n") 'Prepare Prompt Question
    Set oXMLHTTP = CreateObject("MSXML2.ServerXMLHTTP")
    oXMLHTTP.Open "POST", "https://api.openai.com/v1/completions", False 'Set Open Posts For Text
    AIBody = "{" & "model" & ":" & " " & Model & " ", "prompt" & ":" & " " & Prompt & " ", "temperature" & ":" & " " & Temp & " ", "max_tokens" & ":" & " 3500"}" 'Request Body For Text Response
    oXMLHTTP.setRequestHeader "Content-Type", "application/json" 'Set Header Type
    oXMLHTTP.setRequestHeader "Authorization", "Bearer " & SecretKey 'Set Header Authorization
    Debug.Print AIBody
    oXMLHTTP.Send AIBody
    Response = oXMLHTTP.responseText 'Extract Response

    If InStr(Response, "You didn't provide an API key") > 0 Then
        MsgBox "No API Key was added. You can find your API key at " & vbCrLf & "https://platform.openai.com/account/api-keys."
        Exit Function
    End If

    If InStr(Response, "Incorrect API key provided") > 0 Then
        MsgBox "Incorrect API key provided. You can find your API key at " & vbCrLf & "https://platform.openai.com/account/api-keys."
        Exit Function
    End If

    If InStr(Response, "maximum context length") > 0 Then
        MsgBox "You have reached the maximum content length. Please clear the history above to move forward"
        Exit Function
    End If

    Debug.Print Response
    ParsedAnswer = Right(Response, Len(Response) - InStrRev(Response, "{" & Chr(34) & "text" & Chr(34) & ":" & Chr(34)) - 9)
    ParsedAnswer = Left(ParsedAnswer, InStrRev(ParsedAnswer, Chr(34) & "," & Chr(34) & "index" & Chr(34)) - 1)
    ParsedAnswer = Replace(Replace(ParsedAnswer, "\n", Chr(10)), "\" & Chr(34), Chr(34))
    If Left(ParsedAnswer, 2) = Chr(10) & Chr(10) Then ParsedAnswer = Right(ParsedAnswer, Len(ParsedAnswer) - 2)
    ChatGPT = ParsedAnswer
End Function

```

```

1 Option Explicit
2 Dim PromptCell As Range, ParsedAnswer As String
3
4 Sub ReturnDataInTable()
5     Dim oXMLHTTP As Object
6     Dim AIBody As String, Response As String, SecretKey As String, Prompt As String,
7     Temp As String, Model As String
8     Dim StartRow As Long, StartCol As Long, RespRow As Long, RespCol As Long, LastRow As
9     Long, LastCol As Long
10    Dim ColArr() As String, RowArr() As String, RowData As String
11    Dim PromptCell As Range
12    Set PromptCell = ActiveCell
13    Prompt = PromptCell.Value
14    SecretKey = Admin.Range("D5").Value 'Secret Key
15    Model = Admin.Range("D6").Value 'Model
16    Temp = Admin.Range("D7").Value 'Temperature
17    If Model = "" Then Model = "text-davinci-003" 'Set Default Model
18    If Temp = "" Then Temp = "0.2" 'Set Default Temp
19    Prompt = Replace(Replace(Prompt, Chr(34), "\" & Chr(34)), vbNewLine, "\n") & "
20    Represent in a table. Do not include a separator row after first header row."
21    Set oXMLHTTP = CreateObject("MSXML2.ServerXMLHTTP")
22    oXMLHTTP.Open "POST", "https://api.openai.com/v1/completions", False 'Set Open
23    Posts For Text
24    AIBody = "{" "model" ": " "" & Model & " ", " "prompt" ": " "" & Prompt & " ",
25    " "temperature" ": " & Temp & " ", " "max_tokens" ": 3500}" 'Request Body For
26    Text Response
27    oXMLHTTP.setRequestHeader "Content-Type", "application/json" 'Set Header Type
28    oXMLHTTP.setRequestHeader "Authorization", "Bearer " & SecretKey 'Set Header
29    Authorization
30    Debug.Print AIBody
31    On Error GoTo TimeOut
32    oXMLHTTP.Send AIBody
33    Response = oXMLHTTP.responseText 'Extract Response
34    If InStr(Response, "You didn't provide an API key" ) > 0 Then
35        MsgBox "No API Key was added. You can find your API key at " & vbCrLf &
36        "https://platform.openai.com/account/api-keys."
37        Exit Sub
38    End If
39    If InStr(Response, "Incorrect API key provided" ) > 0 Then
40        MsgBox "Incorrect API key provided. You can find your API key at " & vbCrLf &
41        "https://platform.openai.com/account/api-keys."
42        Exit Sub
43    End If
44    If InStr(Response, "maximum context length" ) > 0 Then
45        MsgBox "You have reached the maximum content length. Please clear the history
46        above to move forward"
47        Exit Sub
48    End If
49    Debug.Print Response
50    ParsedAnswer = Right(Response, Len(Response) - InStrRev(Response, "{" & Chr(34) &
51    "text" & Chr(34) & ":" & Chr(34)) - 9)
52    ParsedAnswer = Left(ParsedAnswer, InStrRev(ParsedAnswer, Chr(34) & "," & Chr(34) &
53    "index" ) - 1)
54    ParsedAnswer = Replace(Replace(ParsedAnswer, "\n", Chr(10)), "\" & Chr(34), Chr(34)
55    ))
56    If Left(ParsedAnswer, 2) = Chr(10) & Chr(10) Then ParsedAnswer = Right(ParsedAnswer,
57    Len(ParsedAnswer) - 2)
58    'Parse Response as table
59    StartRow = PromptCell.Row + 1 'Set Initial Response Row 1 row below question
60    StartCol = PromptCell.Column 'Set Initial Start Column
61    RowArr = Split(ParsedAnswer, Chr(10)) 'Split based on new lines

```

1

```
1
48 LastRow = StartRow + UBound(RowArr) 'Last Row
49 For RespRow = StartRow To LastRow
50   RowData = RowArr(RespRow - StartRow) 'Initial Row Data
51   RowData = Left(RowData, Len(RowData) - 1) 'Remove right divider
52   If Left(RowData, 1) = "|" Then RowData = Right(RowData, Len(RowData) - 1)
   'Remove left divider
53   ColArr = Split(RowData, "|") 'Create Array for column data
54   LastCol = StartCol + UBound(ColArr) 'Last Column
55   For RespCol = StartCol To LastCol
56     ActiveSheet.Cells(RespRow, RespCol).Value = ColArr(RespCol - StartCol) 'Add
     Data
57   Next RespCol
58   If RespRow = StartRow Then Range(ActiveSheet.Cells(RespRow, StartCol),
   ActiveSheet.Cells(RespRow, LastCol)).Font.Bold = True 'Bold Header Row
59 Next RespRow
60 ActiveSheet.Range(ActiveSheet.Cells(StartRow, StartCol), ActiveSheet.Cells(LastRow,
   LastCol)).Columns.AutoFit
61 Exit Sub
62 Timeout:
63 MsgBox "The request has timed out. Please try requesting a smaller set of data"
64 End Sub
65
66
```

```

1 Option Explicit
2 Sub FixMyCode()
3     FixMyCodeForm.Show
4     FixMyCodeForm.GenRespTxt.Visible = False
5 End Sub
6 Sub FixMyCode_GetResponse()
7     Dim oXMLHTTP As Object
8     Dim Prompt As String, AIBody As String, Response As String, SecretKey As String,
9     ParsedAnswer As String
10    Dim Temp As String, Model As String, CodePrompt As String
11    CodePrompt = FixMyCodeForm.Prompt.Value
12    If CodePrompt = "" Then
13        MsgBox "Please make sure to enter code or macro that contains the issue. This
14        cannot be left blank"
15        Exit Sub
16    End If
17
18    SecretKey = Admin.Range("D5").Value 'Secret Key
19    Model = Admin.Range("D6").Value 'Model
20    Temp = Admin.Range("D7").Value 'Temperature
21    If Model = "" Then Model = "text-davinci-003" 'Set Default Model
22    If Temp = "" Then Temp = "0.4" 'Set Default Temp
23    Prompt = "Please provide a corrected version of the following ExcelVBA Code and
24    provide an explanation as to the error or issue:" & vbCrLf & CodePrompt 'Build
25    Prompt
26    Prompt = Replace(Replace(Prompt, Chr(34), "\" & Chr(34)), vbCrLf, "\n" )
27    'Prompt Question
28    Set oXMLHTTP = CreateObject("MSXML2.ServerXMLHTTP" )
29    oXMLHTTP.Open "POST", "https://api.openai.com/v1/completions", False 'Set Open
30    Posts For Text
31    AIBody = "{" & "model" & ":" & " " & Model & " ", "prompt" & ":" & " " & Prompt & " ",
32    "temperature" & ":" & " " & Temp & " ", "max_tokens" & ":" & 3500}" 'Request Body For
33    Text Response
34    oXMLHTTP.setRequestHeader "Content-Type", "application/json" 'Set Header Type
35    oXMLHTTP.setRequestHeader "Authorization", "Bearer " & SecretKey 'Set Header
36    Authorization
37    Debug.Print AIBody
38    oXMLHTTP.Send AIBody
39    Response = oXMLHTTP.responseText 'Extract Response
40    If InStr(Response, "You didn't provide an API key" ) > 0 Then
41        MsgBox "No API Key was added. You can find your API key at " & vbCrLf &
42        "https://platform.openai.com/account/api-keys."
43        Exit Sub
44    End If
45    If InStr(Response, "Incorrect API key provided" ) > 0 Then
46        MsgBox "Incorrect API key provided. You can find your API key at " & vbCrLf &
47        "https://platform.openai.com/account/api-keys."
48        Exit Sub
49    End If
50    If InStr(Response, "maximum context length" ) > 0 Then
51        MsgBox "You have reached the maximum content length. Please clear the history
52        above to move forward"
53        Exit Sub
54    End If
55    Debug.Print Response
56    ParsedAnswer = Right(Response, Len(Response) - InStrRev(Response, "{" & Chr(34) &
57    "text" & Chr(34) & ":" & Chr(34)) - 9)
58    ParsedAnswer = Left(ParsedAnswer, InStrRev(ParsedAnswer, Chr(34) & "," & Chr(34) &
59    "index" ) - 1)
60    ParsedAnswer = Replace(Replace(ParsedAnswer, "\n", Chr(10)), "\" & Chr(34), Chr(34)
61    ))

```

1


```
1
47   If Left(ParsedAnswer, 2) = Chr(10) & Chr(10) Then ParsedAnswer = Right(ParsedAnswer,
48       Len(ParsedAnswer) - 2)
49   FixMyCodeForm.Response = ParsedAnswer 'Place Response in userform
50   End Sub
51
52
53   ''''''Some code samples with errors
54
55   Sub PrintSheet()
56       With Sheet1
57           .PageSetup.PrintArea = "$A$1:$D$" & LastRow
58           .PrintOut
59       End With
60   End Sub
61
62   Sub Send_Email()
63       Dim OutApp As Object, OutMail As Object
64       Set OutApp = CreateObject("Application.Outlook" )
65       Set OutMail = OutApp.CreateItem(0)
66       With OutMail
67           .To = "example@example.com"
68           .Subject = "This is the Subject line"
69           .Body = "Hi there, this is the body of the email"
70           .Send
71       End With
72       Set OutMail = Nothing
73       Set OutApp = Nothing
74   End Sub
```

```

1 Option Explicit
2
3 Sub FixMyFormula()
4     Dim oXMLHTTP As Object
5     Dim Prompt As String, AIBody As String, Response As String, SecretKey As String,
        ParsedAnswer As String
6     Dim Temp As String, Model As String, Formula As String
7     Formula = ActiveCell.Formula
8     If Formula = "" Then
9         MsgBox "Please select on a cell with a formula"
10        Exit Sub
11    End If
12    SecretKey = Admin.Range("D5" ).Value 'Secret Key
13    Model = Admin.Range("D6" ).Value 'Model
14    Temp = Admin.Range("D7" ).Value 'Temperature
15    If Model = "" Then Model = "text-davinci-003" 'Set Default Model
16    If Temp = "" Then Temp = "0.4" 'Set Default Temp
17    Prompt = "Please provided a corrected version of the following Excel formula and
        provide an explanation as to the error or issue:" & vbCrLf & Formula 'Build Prompt
18    Prompt = Replace(Replace(Prompt, Chr(34), "\" & Chr(34)), vbCrLf, "\n" )
        'Prompt Question
19    Set oXMLHTTP = CreateObject("MSXML2.ServerXMLHTTP" )
20    oXMLHTTP.Open "POST" , "https://api.openai.com/v1/completions" , False 'Set Open
        Posts For Text
21    AIBody = "{" & "model" & ":" & " " & Model & " " & ", " & "prompt" & ":" & " " & Prompt & " " & ",
        " & "temperature" & ":" & " " & Temp & " " & ", " & "max_tokens" & ":" & "3500"}" 'Request Body For
        Text Response
22    oXMLHTTP.setRequestHeader "Content-Type" , "application/json" 'Set Header Type
23    oXMLHTTP.setRequestHeader "Authorization" , "Bearer " & SecretKey 'Set Header
        Authorization
24    Debug.Print AIBody
25    oXMLHTTP.Send AIBody
26    Response = oXMLHTTP.responseText 'Extract Response
27    If InStr(Response, "You didn't provide an API key" ) > 0 Then
28        MsgBox "No API Key was added. You can find your API key at " & vbCrLf &
            "https://platform.openai.com/account/api-keys."
29        Exit Sub
30    End If
31    If InStr(Response, "Incorrect API key provided" ) > 0 Then
32        MsgBox "Incorrect API key provided. You can find your API key at " & vbCrLf &
            "https://platform.openai.com/account/api-keys."
33        Exit Sub
34    End If
35    If InStr(Response, "maximum context length" ) > 0 Then
36        MsgBox "You have reached the maximum content length. Please clear the history
            above to move forward"
37        Exit Sub
38    End If
39    Debug.Print Response
40    ParsedAnswer = Right(Response, Len(Response) - InStrRev(Response, "[" & Chr(34) &
        "text" & Chr(34) & ":" & Chr(34)) - 9)
41    ParsedAnswer = Left(ParsedAnswer, InStrRev(ParsedAnswer, Chr(34) & "," & Chr(34) &
        "index" ) - 1)
42    ParsedAnswer = Replace(Replace(ParsedAnswer, "\n" , Chr(10)), "\" & Chr(34), Chr(34)
        ))
43    If Left(ParsedAnswer, 2) = Chr(10) & Chr(10) Then ParsedAnswer = Right(ParsedAnswer,
        Len(ParsedAnswer) - 2)
44    With FixMyFormulaForm
45        .Prompt = Prompt
46        .Response = ParsedAnswer

```

1 2

```
47 | 1 2  
48 | | .Show  
49 | | End With  
   | | End Sub
```

A

ActiveCell, 10, 14
 ActiveSheet, 11
 Add, 6
 Admin, 9, 10, 12, 14
 AllBody, 9, 10, 12, 14
 Application, 6
 AutoFit, 11

B

Before, 6
 Body, 13
 Bold, 11

C

Cancel, 6
 Caption, 6
 Cells, 11
 ChatGPT, 9
 Chr, 9, 10, 12-14
 CloseBtn_Click, 7, 8
 cmdBtn, 6
 cmdBtn2, 6
 cmdBtn3, 6
 CodePrompt, 12
 ColArr, 10, 11
 Column, 10
 Columns, 11
 CommandBarButton, 6
 CommandBars, 6
 Controls, 6
 CreateItem, 13
 CreateObject, 9, 10, 12-14

D

Debug, 9, 10, 12, 14
 Delete, 6

E

Explicit, 2-10, 12, 14

F

Faceld, 6
 FixMyCode, 12
 FixMyCode_GetResponse, 7, 12
 FixMyCodeBtn_Click, 7
 FixMyCodeForm, 7, 12, 13
 FixMyFormula, 14
 FixMyFormulaForm, 14
 Font, 11
 Formula, 14

G

GenRespTxt, 12

H

Hide, 7, 8

I

InStr, 9, 10, 12, 14
 InStrRev, 9, 10, 12, 14

L

LastCol, 10, 11
 LastRow, 10, 11, 13
 Left, 9-14
 Len, 9-14

M

Model, 9, 10, 12, 14
 MsgBox, 9-12, 14
 msoControlButton, 6

O

OnAction, 6
 Open, 9, 10, 12, 14
 OutApp, 13
 OutMail, 13
 oXMLHTTP, 9, 10, 12, 14

P

PageSetup, 13
 ParsedAnswer, 9, 10, 12-14
 Print, 9, 10, 12, 14
 PrintArea, 13
 PrintOut, 13
 PrintSheet, 13
 Prompt, 9, 10, 12, 14
 PromptCell, 10

R

Range, 6, 9-12, 14
 Replace, 9, 10, 12, 14
 RespCol, 10, 11
 Response, 9, 10, 12-14
 responseText, 9, 10, 12, 14
 RespData, 10, 11
 ReturnDataInTable, 10
 Right, 9-14
 Row, 10
 RowArr, 10, 11
 RowData, 10, 11

S

SecretKey, 9, 10, 12, 14
 Send, 9, 10, 12-14
 Send_Email, 13
 SetRequestHeader, 9, 10, 12, 14
 Sh, 6
 Sheet1, 13
 Show, 12, 15
 Split, 10, 11
 StartCol, 10, 11
 StartRow, 10, 11
 Subject, 13

T

Target, 6
 Temp, 9, 10, 12, 14
 Temporary, 6
 Timeout, 10, 11

U

UBound, 11

V

Value, 9-12, 14
 vbCrLf, 9, 10, 12, 14
 vbNewLine, 9, 10, 12, 14
 Visible, 12

W

Workbook_Deactivate, 6
 Workbook_SheetBeforeRightClick, 6

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