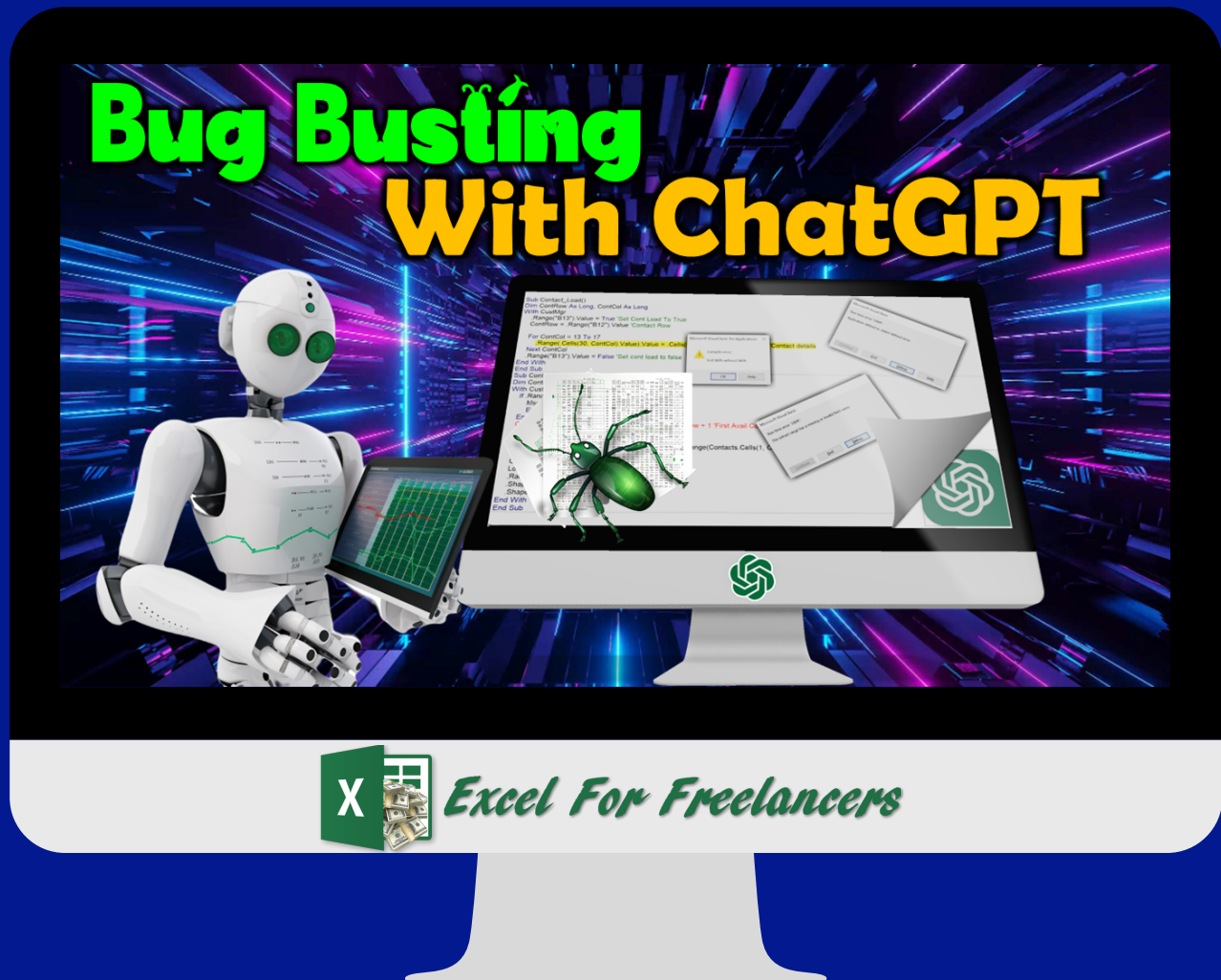


VBA SOURCE CODE BOOK



Can ChatGPT Find & Fix My
Bugs In Excel? Let's Find Out!



[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



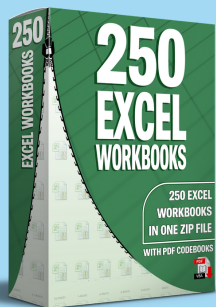
This comprehensive program will take you through a 12-phase process that will turn your enthusiasm for Excel into passive income.

[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](#)



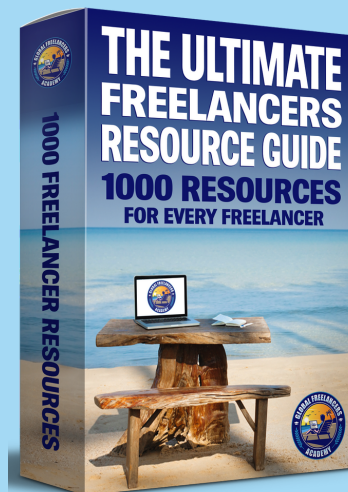
With 1000 live links, continuously updating content, sort-able and filterable items, you will always have exactly what you need, when you need it.

[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
BugFixing.....	2
(Declarations).....	2
Contacts.....	3
(Declarations).....	3
CustMgr.....	4
(Declarations).....	4
Worksheet_Change [Sub].....	4
Worksheet_SelectionChange [Sub].....	4
Customers.....	5
(Declarations).....	5
Invoices.....	6
(Declarations).....	6
ThisWorkbook.....	7
(Declarations).....	7
WorkOrders.....	8
(Declarations).....	8
Modules.....	9
Contact_Macros.....	9
(Declarations).....	9
Contact_AddNew [Sub].....	9
Contact_Delete [Sub].....	9
Contact_Load [Sub].....	9
Contact_SaveNew [Sub].....	9
Cust_LoadMac.....	10
(Declarations).....	10
Customer_Load [Sub].....	10
LoadContacts [Sub].....	10
Cust_MiscMacros.....	11
(Declarations).....	11
Customer_AddNew [Sub].....	11
Customer_CancelNew [Sub].....	11
Customer_Delete [Sub].....	11
Cust_SaveMac.....	12
(Declarations).....	12
Customer_SaveUpdate [Sub].....	12

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("F2" )) Is Nothing And Range("B10" ).Value <> Empty
5         Then Customer_Load
6     If Not Intersect(Target, Range("N4:N12" )) Is Nothing And Range("B13" ).Value =
7         False And Range("B12" ).Value <> Empty Then
8         Dim ContDbRow As Long
9         ContDbRow = Range("R" & Range("B12" ).Value).Value
10        Contacts.Cells(ContDbRow, Target.Offset(0, 1).Value).Value = Target.Value
11        LoadContacts 'Reload Contacts
12    End If
13 End Sub
14
15 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
16     If Not Intersect(Target, Range("M16:N29" )) Is Nothing And Range("M" & Target.Row).
17     Value <> Empty Then
18         Range("B12" ).Value = Target.Row
19         Contact_Load
20     End If
21 End Sub
```


1 Option Explicit

2

1
2

Option Explicit

1 Option Explicit

2

1 Option Explicit

2

```

1  Option Explicit
2
3  Sub Contact_AddNew()
4      With CustMgr
5          .Range("B12,N4:N12" ).ClearContents 'Clear Existing Contact
6          .Shapes("ExistContGrp" ).Visible = msoFalse
7          .Shapes("SaveContBtn" ).Visible = msoCTrue
8      End With
9  End Sub
10
11 Sub Contact_Delete()
12     Dim ContRow As Long, SelRow As LoadPictureConstants
13     With CustMgr
14         If .Range("B12" ).Value = Empty Then Exit Sub
15         SelRow = .Range("B12" ).Value 'Selected Row
16         ContRow = .Range("R" & SelRow).Value 'Contact Row
17         If ContRow <> Empty Then Contacts.Range(ContRow & ":" & ContRow).EntireRow
18         LoadContacts
19         If .Range("M16" ).Value <> Empty Then .Range("M16" ).Select Else: Contact_AddNew
20     End With
21 End Sub
22
23
24 Sub Contact_Load()
25     Dim ContRow As Long, ContCol As Long
26     With CustMgr
27         .Range("B13" ).Value = True 'Set Cont Load To True
28         ContRow = .Range("B12" ).Value 'Contact Row
29
30         For ContCol = 13 To 17
31             .Range(.Cells(30, ContCol).Value).Value = .Cells(ContRow, ContCol).Value
32             'Contact details
33         Next ContCol 'Added missing Next statement
34         .Range("B13" ).Value = False 'Set cont load to false
35     End With
36 End Sub
37
38 Sub Contact_SaveNew()
39     Dim ContDbRow As Long, ContDbCol As Long
40     With CustMgr
41         If .Range("N4" ).Value = Empty Then
42             MsgBox "Please enter a Contact First Name"
43             Exit Sub
44         End If
45         ContDbRow = Contacts.Range("A99999" ).End(xlUp).Row + 1 'First Avail Cont Row
46
47         For ContDbCol = 1 To 6
48             Contacts.Cells(ContDbRow, ContDbCol).Value = .Range(Contacts.Cells(1,
49             ContDbCol).Value).Value
50         Next ContDbCol
51         Contacts.Range("G" & ContDbRow).Value = "=Row()"
52         LoadContacts
53         .Range("B12" ).Value = .Range("M29" ).End(xlUp).Row
54         .Shapes("ExistContGrp" ).Visible = msoCTrue
55         .Shapes("SaveContBtn" ).Visible = msoFalse
56     End With
57 End Sub

```

```

1 Option Explicit
2
3 Sub Customer_Load()
4     Dim CustRow As Long, LastInvRow As Long, LastWORow As Long, CustCol As Long
5     With CustMgr
6         If .Range("B10").Value = Empty Then
7             MsgBox "Please select a correct customer name from the list"
8             Exit Sub
9         End If
10        CustRow = .Range("B10").Value
11
12        'Clear Existing Data
13        .Range("B12,E4:G8,I4,K4,I6:K6,E8:G8,I8:K8,E10:K10,E12:G12,I12,K12,D14:K16").
14        ClearContents 'Clear Customer Cells
15        .Range("D20:G29,I20:K29,M16:N29,N4:N12").ClearContents 'Clear All Other Fields
16
17        For CustCol = 1 To 13
18            .Range(Customers.Cells(1, CustCol).Value).Value = Customers.Cells(CustRow,
19            CustCol).Value
20        Next CustCol
21        .Shapes("ExistCustGrp").Visible = msoCTrue
22        .Shapes("NewCustGrp").Visible = msoFalse
23        .Shapes("SaveContBtn").Visible = msoFalse
24        .Shapes("ExistContGrp").Visible = msoCTrue
25
26        LoadContacts 'Load Contacts
27
28        'Load Invoice
29        LastInvRow = Invoices.Range("A99999").End(xlUp).Row
30        Invoices.Range("I3:L99999").ClearContents 'Clear any previous results
31        Invoices.Range("A2:E" & LastInvRow).AdvancedFilter xlFilterCopy, CriteriaRange:=
32        Invoices.Range("G2:G3"), CopyToRange:=Invoices.Range("I2:L2"), Unique:=False
33        .Range("D20:G29").Value = Invoices.Range("I3:L12").Value 'Bring in all Invoice
34        data
35
36        'Load Work Orders
37        LastWORow = WorkOrders.Range("A99999").End(xlUp).Row
38        WorkOrders.Range("H3:J99999").ClearContents 'Clear any previous results
39        WorkOrders.Range("A2:D" & LastWORow).AdvancedFilter xlFilterCopy, CriteriaRange:=
40        =WorkOrders.Range("F2:F3"), CopyToRange:=WorkOrders.Range("H2:J2"), Unique:=
41        False
42        .Range("I20:K29").Value = WorkOrders.Range("H3:J12").Value 'Bring in all WO
43        Data
44    End With
45 End Sub
46
47 Sub LoadContacts()
48     Dim LastContRow As Long
49     LastContRow = Contacts.Range("A99999").End(xlUp).Row
50     Contacts.Range("L3:Q999").ClearContents 'Clear any previous results
51     Contacts.Range("A3:G" & LastContRow).AdvancedFilter xlFilterCopy, Contacts.Range(
52     "J2:J3"), CopyToRange:=Contacts.Range("L2:Q2"), Unique:=False
53     CustMgr.Range("M16:R29").Value = Contacts.Range("L3:Q16").Value 'Copy over
54     any/all contacts
55 End Sub

```

```
1 Option Explicit
2
3 Sub Customer_AddNew()
4     With CustMgr
5         .Range("B6" ).Value = True 'Set new customer to true
6         .Shapes("ExistCustGrp" ).Visible = msoFalse
7         .Shapes("NewCustGrp" ).Visible = msoCTrue
8         'Clear Existing Data
9         .Range(
10            "B2,F2:G2,B12,E4:G8,I4,K4,I6:K6,E8:G8,I8:K8,E10:K10,E12:G12,I12,K12,D14:K16" ).
11            ClearContents 'Clear Customer Cells
12        End With
13    End Sub
14
15 Sub Customer_CancelNew()
16     If Customers.Range("B4" ).Value < Empty Then CustMgr.Range("F2" ).Value = Customers.
17     Range("B4" ).Value
18 End Sub
19
20 Sub Customer_Delete()
21     Dim CustRow As Long
22     If CustMgr.Range("B4" ).Value <> Empty Then
23         CustRow = CustMgr.Range("B4" ).Value 'Customer Row
24         Customers.Range(CustRow & ":" & CustRow).EntireRow.Delete
25         Customer_CancelNew
26     End If
27 End Sub
```

```
1 Option Explicit
2
3 Sub Customer_SaveUpdate()
4     Dim CustRow As Long, CustCol As Long
5     With CustMgr
6         If .Range("E4" ).Value = Empty Then
7             MsgBox "Please enter a customer name"
8             Exit Sub
9         End If
10        'Determine if new or existing customer
11        If .Range("B4" ).Value = Empty Then 'New customer
12            CustRow = Customers.Range("A99999" ).End(xlUp).Row + 1 'First Avail Customer Row
13            .Range("B2" ).Value = .Range("B8" ).Value
14            Customers.Range("A" & CustRow).Value = .Range("B8" ).Value 'Next Cust ID #
15        Else 'Existing customer
16            CustRow = .Range("B4" ).Value 'Customer Row
17        End If
18
19        For CustCol = 2 To 13
20            Customers.Cells(CustRow, CustCol).Value = .Range(Customers.Cells(1, CustCol).
                Value).Value
21        Next CustCol
22        .Shapes("ExistCustGrp" ).Visible = msoCTrue
23        .Shapes("NewCustGrp" ).Visible = msoFalse
24        .Shapes("SaveContBtn" ).Visible = msoFalse
25        .Shapes("ExistContGrp" ).Visible = msoCTrue
26        .Range("B6" ).Value = False 'Existing customer
27    End With
28 End Sub
29
```


A

AdvancedFilter, 10

C

Cells, 4, 9, 10, 12

ClearContents, 9-11

Contact_AddNew, 9

Contact_Delete, 9

Contact_Load, 4, 9

Contact_SaveNew, 9

Contacts, 4, 9, 10

ContCol, 9

ContDbCol, 9

ContDbRow, 4, 9

ContRow, 9

CopyToRange, 10

CriteriaRange, 10

CustCol, 10, 12

CustMgr, 9-12

Customer_AddNew, 11

Customer_CancelNew, 11

Customer_Delete, 11

Customer_Load, 4, 10

Customer_SaveUpdate, 12

Customers, 10-12

CustRow, 10-12

D

Delete, 11

E

Empty, 4, 9-12

EntireRow, 9, 11

Explicit, 2-12

I

Intersect, 4

Invoices, 10

L

LastContRow, 10

LastInvRow, 10

LastWORow, 10

LoadContacts, 4, 9, 10

LoadPictureConstants, 9

M

MsgBox, 9, 10, 12

msoCTrue, 9-12

msoFalse, 9-12

O

Offset, 4

R

Range, 4, 9-12

Row, 4, 9, 10, 12

S

SelRow, 9

Shapes, 9-12

T

Target, 4

U

Unique, 10

V

Value, 4, 9-12

Visible, 9-12

W

WorkOrders, 10

Worksheet_Change, 4

Worksheet_SelectionChange, 4

X

xlFilterCopy, 10

xlUp, 9, 10, 12

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