

# VBA SOURCE CODE BOOK

## WEBSITE ORDER SYNC UPDATED

**EXCLUSIVE FOR  
SILVER & GOLD  
MEMBERS**

**FEATURE, FIX,  
OR FOCUS**



*Excel For Freelancers*

**How To Create An E-Commerce  
Website, Sync Orders To Excel  
AND Create A Dynamic  
Dashboard**



**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 300,000 YouTube subscribers, 15,000,000 video views, 200+ comprehensive training videos, and a thriving 50,000 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



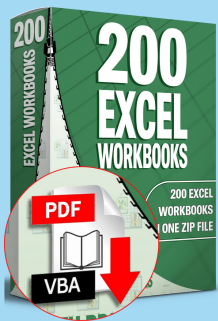
This comprehensive program will take you through a 12-phase process that will turn your enthusiasm for Excel into passive income.

[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 200 of my BEST Applications now with PDF VBA Codebooks packed into a SINGLE ZIP File which also includes the "200 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](#)

## Table of Contents

Projects.....	2
VBAProject.....	2
Documents.....	2
Admin.....	2
(Declarations).....	2
CustOrders.....	3
(Declarations).....	3
Worksheet_SelectionChange [Sub ].....	3
Dashboard.....	4
(Declarations).....	4
OrderItems.....	5
(Declarations).....	5
Orders.....	6
(Declarations).....	6
Worksheet_Change [Sub ].....	6
Worksheet_SelectionChange [Sub ].....	6
PivotData.....	7
(Declarations).....	7
Products.....	8
(Declarations).....	8
Sheet2.....	9
(Declarations).....	9
ThisWorkbook.....	10
(Declarations).....	10
Workbook_Open [Sub ].....	10
Forms.....	11
AddProdForm.....	11
(Declarations).....	11
AddProdPicBtn_Click [Sub ].....	11
CancelBtn_Click [Sub ].....	11
SaveBtn_Click [Sub ].....	11
Modules.....	13
App_Macros.....	13
(Declarations).....	13
AddNewProduct [Sub ].....	13
BrowseForDataFolder [Sub ].....	13
BrowseForPicFolder [Sub ].....	13
BrowseForProdPicture [Sub ].....	13
Order_Macros.....	14
(Declarations).....	14
Order_Load [Sub ].....	14
Order_Print [Sub ].....	14
Order_SaveAsPDF [Sub ].....	14
ViewOrder [Sub ].....	14
OrderData_Macros.....	15
(Declarations).....	15
CheckForNewOrderData [Sub ].....	15

1	Option Explicit
2	

```
1 Option Explicit
2
3 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
4     If Target.CountLarge > 1 Then Exit Sub
5     If Not Intersect(Target, Range("A3:J99999" )) Is Nothing And Range("A" & Target.Row
6     ).Value <> Empty Then Range("AA1" ).Value = Target.Row
7 End Sub
```

1  
2

Option Explicit

1 Option Explicit

2



```
1 Option Explicit
2
3 Private Sub Worksheet_Change(ByVal Target As Range)
4     If Target.CountLarge > 1 Then Exit Sub
5     If Not Intersect(Target, Range("D1" )) Is Nothing Then Order_Load
6 End Sub
7
8 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
9
10 End Sub
```

1 Option Explicit

2

1  
2

Option Explicit

1  
2

Option Explicit

```
1 Option Explicit
2
3 Private Sub Workbook_Open()
4     'CheckForNewOrderData
5 End Sub
```

```

1
2 Option Explicit
3
4
5 Private Sub AddProdPicBtn_Click()
6     BrowseForProdPicture
7 End Sub
8
9 Private Sub CancelBtn_Click()
10    AddProdForm.Hide
11 End Sub
12
13 Private Sub SaveBtn_Click()
14    Dim ProductFld As Control
15    Dim ProdRow As Long, ProdCol As Long, CatRow As Long
16    Dim FileName As String, URL As String, Json As String, Response As String, CatID As
17    String
18    Dim ObjHTTP As Object
19    Dim ProdData As String
20    If Me.Field1.Value = Empty Then
21        MsgBox "Please make sure to save a Product Name before saving"
22        Exit Sub
23    End If
24    With Products
25        ProdRow = .Range("A99999" ).End(xlUp).Row + 1 'First avail Row
26        On Error Resume Next
27        .Range("A" & ProdRow).Value = Application.WorksheetFunction.Max(.Range(
28            "Prod_ID" )) + 1
29        On Error GoTo 0
30        If .Range("A" & ProdRow).Value = Empty Then .Range("A" & ProdRow).Value = 1
31        'Set First Prod ID to 1
32        ProdData = "Field1=" & .Range("A" & ProdRow).Value 'Set Product ID
33        For ProdCol = 2 To 7
34            Set ProductFld = Me.Controls("Field" & ProdCol - 1)
35            .Cells(ProdRow, ProdCol).Value = ProductFld.Value
36            If ProdCol = 4 Then 'Set Category
37                On Error Resume Next
38                CatRow = Admin.Range("Categories" ).Find(ProductFld.Value, , xlValues,
39                    xlWhole).Row
40                On Error GoTo 0
41                If CatRow <> 0 Then ProdData = ProdData & "&Field" & ProdCol & "=" &
42                    Admin.Range("C" & CatRow).Value 'Set Category ID
43            Else 'Everything But Category
44                ProdData = ProdData & "&Field" & ProdCol & "=" & ProductFld.Value 'Build
45                Data String
46            End If
47        Next ProdCol
48        AddProdForm.Hide
49
50        Set ObjHTTP = CreateObject("MSXML2.ServerXMLHTTP" )
51        URL = "https://hook.integromat.com/dtrwyeo2o8rhsi649aq5add6t9kv48gh?" & ProdData
52        'Send Information via webhook
53        ObjHTTP.Open "POST" , URL, False
54        ObjHTTP.setRequestHeader "Content-type" , "application/json"
55        'On Error GoTo NoInternet
56        ObjHTTP.Send (Json) 'Send Information
57        Response = ObjHTTP.responsetext
58        If Response <> "Accepted" Then
59            MsgBox "Details not accepted. Please check your webhook & variables for
60                accuracy"
61            Exit Sub
62        End If
63    End With
64 End Sub

```

1 2 3

```
54 1 2 3  
55 | | | End If  
56 | | End With  
57 | End Sub  
58
```

```

1 Option Explicit
2 Sub AddNewProduct()
3     AddProdForm.Show
4 End Sub
5
6
7
8 Sub BrowseForDataFolder()
9     Dim AppFolder As FileDialog
10    Set AppFolder = Application.FileDialog(msoFileDialogFolderPicker)
11    With AppFolder
12        .Title = "Browse For Order Data Folder"
13        .AllowMultiSelect = False
14        If .Show <> -1 Then GoTo NoSelection
15        Admin.Range("C3" ).Value = .SelectedItems(1)
16    End With
17    NoSelection:
18 End Sub
19
20
21 Sub BrowseForPicFolder()
22     Dim AppFolder As FileDialog
23     Set AppFolder = Application.FileDialog(msoFileDialogFolderPicker)
24     With AppFolder
25         .Title = "Browse For Product Pictures Folder"
26         .AllowMultiSelect = False
27         If .Show <> -1 Then GoTo NoSelection
28         Admin.Range("C4" ).Value = .SelectedItems(1)
29     End With
30     NoSelection:
31 End Sub
32
33 Sub BrowseForProdPicture()
34     Dim ProdFolder As String, PicName As String
35     Dim ProdPic As FileDialog
36     If Admin.Range("C4" ).Value = Empty Or Dir(Admin.Range("C4" ).Value, vbDirectory) =
37         "" Then
38         MsgBox "Please select a folder for all Product Pictures in the Admin screen"
39         Exit Sub
40     End If
41     ProdFolder = Admin.Range("C4" ).Value & "\" 'Step Picture Folder
42     Set ProdPic = Application.FileDialog(msoFileDialogFilePicker)
43     With ProdPic
44         .Title = "Please select a Product picture"
45         .Filters.Add "Picture Files" , "*.jpg,*.png,*.gif" , 1
46         .AllowMultiSelect = False
47         If .Show <> -1 Then GoTo NoSelection
48         FileCopy .SelectedItems(1), ProdFolder & Dir(.SelectedItems(1)) 'Copy Recipe
49         Picture into folder
50         PicName = Dir(.SelectedItems(1)) 'Set Picture Name
51         AddProdForm.Field6.Value = PicName ' Add Pic Name To Field
52     End With
53     NoSelection:
54 End Sub

```



```

1 Option Explicit
2 Dim OrderRow As Long, LastRow As Long, LastResultRow As Long, ResultRow As Long,
  OrdItemRow As Long
3
4 Sub Order_Load()
5     With Orders
6         If .Range("B4").Value = Empty Then
7             MsgBox "Please enter a correct Order #"
8             Exit Sub
9         End If
10
11         OrderRow = .Range("B4").Value 'Order Row
12         .Range("I1").Value = .Range("D1").Value 'Set Order ID
13         .Range("D1,F5:F7,I3,I5,E9:I31").ClearContents 'Clear Search Order
14         .Range("I3").Value = CustOrders.Range("B" & OrderRow).Value 'Date
15         .Range("F5").Value = CustOrders.Range("C" & OrderRow).Value 'Customer Name
16         .Range("F6").Value = CustOrders.Range("D" & OrderRow).Value 'Customer Address
17         .Range("F7").Value = CustOrders.Range("E" & OrderRow).Value & ". " _
18         & CustOrders.Range("F" & OrderRow).Value & " " _
19         & CustOrders.Range("G" & OrderRow).Value 'City, State Zip
20         .Range("I5").Value = CustOrders.Range("I" & OrderRow).Value 'Payment Type
21
22         'Load Order Items
23         With OrderItems
24             LastRow = .Range("A9999").End(xlUp).Row 'Last Item Row
25             If LastRow < 3 Then GoTo NoItems
26             .Range("A2:J" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
27                 "N1:N2"), CopyToRange:=.Range("Q2:V2"), Unique:=False
28             LastResultRow = .Range("Q9999").End(xlUp).Row
29             If LastResultRow < 3 Then GoTo NoItems
30             For ResultRow = 3 To LastResultRow
31                 OrdItemRow = .Range("V" & ResultRow).Value 'Order Row
32                 Orders.Range("E" & OrdItemRow & ":I" & OrdItemRow).Value = .Range("Q" &
33                     ResultRow & ":V" & ResultRow).Value 'Item Details
34             Next ResultRow
35         End With
36         NoItems:
37     End With
38 End Sub
39
40 Sub Order_Print()
41     Orders.PrintOut , , , False, , , , False
42 End Sub
43
44 Sub Order_SaveAsPDF()
45     Dim FilePath As String
46     FilePath = ThisWorkbook.Path & "\" & Orders.Range("F5").Value & "_" & Orders.
47     Range("I1").Value 'File Path 'Set customer Name & Inv. ID
48     If Dir(FilePath, vbDirectory) <> "" Then Kill (FilePath)
49     Orders.ExportAsFixedFormat xlTypePDF, FilePath, , , False, , , True
50 End Sub
51
52 Sub ViewOrder()
53     If CustOrders.Range("AA1").Value = Empty Then Exit Sub
54     Orders.Range("D1").Value = CustOrders.Range("A" & CustOrders.Range("AA1").Value).
55     Value
56     Orders.Activate
57 End Sub

```

```

1 Option Explicit
2
3 Sub CheckForNewOrderData()
4     Dim OrderFolder As String, DataFilePath As String, FileName As String, OrderData As String
5     Dim MainOrdData As String, OrdItemData As String, OrderID As String, RowData As String, ProdID As String
6     Dim OrderRow As Long, OrderCol As Long, OrderCount As Long, ProdRow As Long
7     Dim OrderItemDBRow As Long, OrdItemCol As Long, OrdItem As Long, OrderItemRow As Long
8     Dim OrderDataArr() As String, MainOrderArr() As String, OrderItemArr() As String, RowDataArr() As String
9     Dim OrderDt As Date
10
11     OrderFolder = Admin.Range("DataFolder").Value & "\"
12     If OrderFolder = Empty Or Dir(OrderFolder, vbDirectory) = "" Then
13         MsgBox "Please make sure to browse for a correct Order Data Folder"
14         BrowseForDataFolder 'Run Macro to browse for folder
15         If OrderFolder = Empty Or Dir(OrderFolder, vbDirectory) = "" Then Exit Sub
16         'Exit on 2nd attempt if incorrect path
17     End If
18     With CustOrders
19         FileName = Dir(OrderFolder & "*.txt") 'Start Loop
20         Do While Len(FileName) > 0
21             DataFilePath = OrderFolder & FileName 'Complete File Path of Text file
22             Open DataFilePath For Input As #1
23             Line Input #1, OrderData 'Create String of data
24             Close #1
25             OrderDataArr = Split(OrderData, "/OrdItems/") 'Split string based on Main
26             Order Details and Order Items
27             MainOrdData = OrderDataArr(0) 'Main Order Data
28             OrdItemData = OrderDataArr(1) 'Order Items Data (individual Order
29             Items details)
30             MainOrderArr = Split(MainOrdData, "/c/") 'Split Main Order Data
31             OrderItemArr = Split(OrdItemData, "/ir/") 'Split Order Items Data By Rows
32             Rows
33             'Add in main Order data
34             OrderID = MainOrderArr(0) 'Extract Order ID
35             OrderDt = MainOrderArr(1) 'Set Order Date Variable
36             On Error Resume Next
37             OrderRow = .Range("Order_ID").Find(OrderID, , xlValues, xlWhole).Row
38             'Determind Existing Row (if any)
39             On Error GoTo 0
40             If OrderRow = 0 Then OrderRow = .Range("A99999").End(xlUp).Row + 1 'first
41             Avail Row
42             For OrderCol = 1 To 10
43                 .Cells(OrderRow, OrderCol).Value = MainOrderArr(OrderCol - 1) 'Add Main
44                 Order data Order Database
45             Next OrderCol
46             'Add in Table Data
47             OrderItemDBRow = OrderItems.Range("A99999").End(xlUp).Row + 1 'first
48             Available Row Items DB
49             OrderItemRow = 9 'Set Intial Order Item Row
50             For OrdItem = LBound(OrderItemArr) To UBound(OrderItemArr) 'Loop Through
51                 Order Rows
52                 RowData = OrderItemArr(OrdItem)
53                 RowDataArr = Split(RowData, "/ic/")
54                 If RowDataArr(0) = "" Then GoTo NextFile

```

1 2 3 4

```
49      1 2 3 4      OrderItems.Range("A" & OrderItemDBRow).Value = OrderID 'Order ID
50      OrderItems.Range("B" & OrderItemDBRow).Value = OrderDt 'Order Date
51      For OrdItemCol = 3 To 7
52          OrderItems.Cells(OrderItemDBRow, OrdItemCol).Value = RowDataArr(
                    OrdItemCol - 3) 'Place Table Data
53      Next OrdItemCol
54      OrderItems.Range("H" & OrderItemDBRow).Value = OrderItemRow 'Order Row
55      ProdID = RowDataArr(0) 'Set Product ID
56      On Error Resume Next
57      ProdRow = Products.Range("Prod_ID").Find(ProdID, , xlValues, xlWhole).Row
                    'Get Product Row
58      On Error GoTo 0
59      If ProdRow <> 0 Then OrderItems.Range("I" & OrderItemDBRow).Value =
                    Products.Range("D" & ProdRow).Value 'Set Category
60      OrderItemDBRow = OrderItemDBRow + 1 'Increment Order Item DB Row
61      OrderItemRow = OrderItemRow + 1 'Increment Order Item Row
62  Next OrdItem
63      OrderRow = OrderRow + 1 'Increment Order Row
64      NextFile:
65      FileName = Dir() 'Clears out current file name
66      Kill (DataFilePath)
67      OrderCount = OrderCount + 1 'Count Text Files Converted
68  Loop
69      MsgBox OrderCount & " Orders have been added"
70  End With
71 End Sub
72
```

—  
\_, 14

## A

Activate, 14  
Add, 13  
AddNewProduct, 13  
AddProdForm, 11, 13  
AddProdPicBtn\_Click, 11  
Admin, 11, 13, 15  
AdvancedFilter, 14  
AllowMultiSelect, 13  
AppFolder, 13  
Application, 11, 13

## B

BrowseForDataFolder, 13, 15  
BrowseForPicFolder, 13  
BrowseForProdPicture, 11, 13

## C

CancelBtn\_Click, 11  
CatID, 11  
CatRow, 11  
Cells, 11, 15, 16  
CheckForNewOrderData, 15  
ClearContents, 14  
Close, 15  
Control, 11  
Controls, 11  
CopyToRange, 14  
CountLarge, 3, 6  
CreateObject, 11  
CriteriaRange, 14  
CustOrders, 14, 15

## D

DataFilePath, 15, 16  
Dir, 13-16

## E

Empty, 3, 11, 13-15  
Explicit, 2-11, 13-15  
ExportAsFixedFormat, 14

## F

Field1, 11  
Field6, 13  
FileCopy, 13  
FileDialog, 13  
FileName, 11, 15, 16  
FilePath, 14  
Filters, 13  
Find, 11, 15, 16

## H

Hide, 11

## I

Input, 15  
Intersect, 3, 6

## J

Json, 11

## K

Kill, 14, 16

## L

LastResultRow, 14  
LastRow, 14  
LBound, 15  
Len, 15  
Line, 15

## M

MainOrdData, 15  
MainOrderArr, 15  
Max, 11  
MsgBox, 11, 13-16  
msoFileDialogFilePicker, 13  
msoFileDialogFolderPicker, 13

## N

NextFile, 15, 16  
NoItems, 14  
NoSelection, 13

## O

ObjHTTP, 11  
Open, 11, 15  
Order\_Load, 6, 14  
Order\_Print, 14  
Order\_SaveAsPDF, 14  
OrderCol, 15  
OrderCount, 15, 16  
OrderData, 15  
OrderDataArr, 15  
OrderDt, 15, 16  
OrderFolder, 15  
OrderID, 15, 16  
OrderItemArr, 15  
OrderItemDBRow, 15, 16  
OrderItemRow, 15, 16  
OrderItems, 14-16  
OrderRow, 14-16  
Orders, 14  
OrdItem, 15, 16  
OrdItemCol, 15, 16  
OrdItemData, 15  
OrdItemRow, 14

## P

Path, 14  
PicName, 13  
PrintOut, 14  
ProdCol, 11  
ProdData, 11  
ProdFolder, 13  
ProdID, 15, 16  
ProdPic, 13  
ProdRow, 11, 15, 16  
ProductFld, 11  
Products, 11, 16

## R

Range, 3, 6, 11, 13-16  
Response, 11  
responsetext, 11  
ResultRow, 14  
Row, 3, 11, 14-16  
RowData, 15  
RowDataArr, 15, 16

## S

SaveBtn\_Click, 11  
SelectedItems, 13

Send, 11  
setRequestHeader, 11  
Show, 13  
Split, 15

## T

Target, 3, 6  
ThisWorkbook, 14  
Title, 13

## U

UBound, 15  
Unique, 14  
URL, 11

## V

Value, 3, 11, 13-16  
vbDirectory, 13-15  
ViewOrder, 14

## W

Workbook\_Open, 10  
Worksheet\_Change, 6  
Worksheet\_SelectionChange, 3, 6  
WorksheetFunction, 11

## X

xlFilterCopy, 14  
xlTypePDF, 14  
xlUp, 11, 14, 15  
xlValues, 11, 15, 16  
xlWhole, 11, 15, 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*