

VBA SOURCE CODE BOOK

BILL OF MATERIALS (BOM)

General Info | Assembly Items | Select Item | Add New | Update | Delete Item

Item Name: Full Computer Set | Item Type: Assembly | Part/Serial #: FS4584

Purchase Description: Lenovo Complete SD27 | Sales Description: Lenovo Full Computer Assembly | Unit Of Measure: each

Def. Purch. Qty: 10 | Def. Sales Qty: 1 | Category: Assembly

Item Cost: \$584.99 | Sales Price: \$845.99 | Qty On Hand: 12

Item Notes: 30 Day Warranty

1-Motherboard | 1-Computer Fan 120 | 1-Graphics Card 16G | 1-Hard Drive 1TB SSD | 1-Keyboard Wireless | 1-Monitor 27" Black | 1-Mouse Wireless | 1-Power Supply

Item Name	Item Desc.	Qty Required	Item Cost	Sales Price	Qty In Stock
Motherboard 32G	Motherboard 32G Gr	1	\$89.00	\$149.00	8
Computer Fan 120	Computer Fan 120 Vc	1	\$25.00	\$35.00	8
Graphics Card 16G	Graphics Card 16GB C	1	\$78.00	\$109.00	8
Hard Drive 1TB SSD	Hard Drive 1TB SSD C	1	\$68.00	\$98.00	8
Keyboard Wireless	Keyboard Wireless -	1	\$14.00	\$29.00	8
Monitor 27" Black	Monitor 27" Black - T	1	\$99.00	\$149.00	5
Mouse Wireless	Mouse Wireless - Qu	1	\$14.99	\$19.99	8
Power Supply	Power Supply 120V V	1	\$45.00	\$65.00	8
RAM 8G DDR	RAM 8G DDR Fast	2	\$48.00	\$58.00	6
Tower Case - Mini	Tower Case - Mini P	1	\$56.00	\$76.00	8

ASSEMBLY SUMMARY

Total Assemb. Cost	\$ 584.99
Total Sales Price	\$ 845.99
Total Assembled	12
Available to Build	1
Assemble Quantity	
Set To Disassemble Qty	

Excel For Freelancers

How To Create An Inventory Assembly & Bill Of Materials (BOM) Application 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 nearly 300,000 YouTube subscribers, 14,000,000 video views, 200+ comprehensive training videos, and a thriving 40,000 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



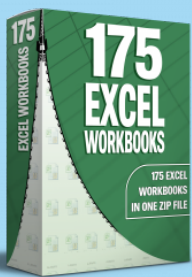
This comprehensive program will take you through a 12-phase process that will turn your enthusiasm for Excel into passive income.

[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 175 of my BEST Applications into a SINGLE ZIP File which also includes the "175 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](#)

Projects.....	2
VBAProject	2
Documents	2
Admin.....	2
(Declarations).....	2
AssembDB	3
(Declarations).....	3
Customers.....	4
(Declarations).....	4
ItemDB	5
(Declarations).....	5
Items	6
(Declarations).....	6
Worksheet_Change [Sub].....	6
Worksheet_SelectionChange [Sub]	6
OrderItems	8
(Declarations).....	8
OrderList	9
(Declarations).....	9
Orders	10
(Declarations).....	10
Worksheet_Change [Sub]	10
ThisWorkbook	12
(Declarations).....	12
Vendors.....	13
(Declarations).....	13
Modules.....	14
Assembly_macros	14
(Declarations).....	14
Assembly_Build [Sub]	14
Assembly_Disassemble [Sub]	14
Assembly_Expand [Sub]	15
Assembly_Shrink [Sub]	16
Item_Macros	17
(Declarations).....	17
Item_AddItemPic [Sub]	17
Item_AddNew [Sub]	17
Item_Cancel_New [Sub].....	17
Item_ClearItemPic [Sub]	17
Item_Delete [Sub]	17
Item_Load [Sub].....	18
Item_SaveUpdate [Sub]	18
Item_ShowPic [Sub].....	19
OrderMacros	21
(Declarations).....	21
Order_CancelNew [Sub]	21
Order_Load [Sub]	21
Order_New [Sub]	21
Order_Print [Sub]	21
Order_SaveUpdate [Sub]	21

1	Option Explicit
2	

1	Option Explicit
2	

1	Option Explicit
2	

1	Option Explicit
2	

Option Explicit

```

1 Private Sub Worksheet_Change(ByVal Target As Range)
2
3   Dim FoundItem As Range
4   Dim ItemRow As Long
5   Dim ItemNm As String
6   If Target.CountLarge > 1 Then Exit Sub
7   If Not Intersect(Target, Range("I3" )) Is Nothing And Range("I3" ).Value <> Empty Then
8     Range("D3" ).Select 'Set to Gen. Info Tab
9     Range("B3" ).Value = Range("B2" ).Value 'Item ID
10    Item_Load
11  End If
12
13
14  'On assembly item change
15  If Not Intersect(Target, Range("E36:E58" )) Is Nothing And Range("B6" ).Value = False Then
16
17    If Target.Value <> Empty Then
18      ItemNm = Target.Value
19      Set FoundItem = ItemDB.Range("Item_Name" ).Find(ItemNm, , xlValues, xlWhole)
20      If Not FoundItem Is Nothing Then 'Item found
21        ItemRow = FoundItem.Row
22        Range("F" & Target.Row).Value = ItemDB.Range("G" & ItemRow).Value 'Item
23        Description
24        Range("G" & Target.Row).Value = 1 'Set Default Qty to 1
25        Range("H" & Target.Row).Value = ItemDB.Range("K" & ItemRow).Value 'Item cost
26        Range("I" & Target.Row).Value = ItemDB.Range("L" & ItemRow).Value 'Sales Price
27
28      Else
29        MsgBox "Item not found"
30      End If
31    Else 'Cleared
32      Range("F" & Target.Row & ":I" & Target.Row).ClearContents
33    End If
34    'Update Default Assembly Costs & Sales Prices
35    Range("F11" ).Value = Range("M35" ).Value 'Update Assembly Cost
36    Range("H11" ).Value = Range("M36" ).Value 'Update Assembly Sales Price
37  End If
38 End Sub

```

```

39 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
40   If Target.CountLarge > 2 Then Exit Sub
41   If Not Intersect(Target, Range("D3:F3" )) Is Nothing And Range("H5" ).Value = "Assembly"
42   Then
43     Range("B1" ).Value = Target.Column 'target column
44     'Hide Everything
45     Application.ScreenUpdating = False
46     Shapes("PicBtnGrp" ).Visible = msoFalse
47     Shapes("AssembGrp" ).Visible = msoFalse
48     On Error Resume Next
49     Shapes("ItemPic" ).Visible = msoFalse
50     On Error GoTo 0
51     Range("4:60" ).EntireRow.Hidden = True
52     If Target.Column = 4 Then 'General Info
53       Range("4:31" ).EntireRow.Hidden = False 'Display Rows
54       Shapes("PicBtnGrp" ).Visible = msoCTrue
55       On Error Resume Next
56       Shapes("ItemPic" ).Visible = msoCTrue
57       On Error GoTo 0
58       If Range("H5" ).Value = "Assembly" Then Shapes("ExpandBtn" ).Visible = msoCTrue
59       Range("F5" ).Select

```

1 2 3

```
58     1 2 3 Else ' Assembly
59     Range("33:60").EntireRow.Hidden = False 'Display Rows
60     Shapes("AssembGrp").Visible = msoCTrue
61     Range("E36").Select
62     If Range("H5").Value = "Assembly" Then
63         If Shapes("ShrinkBtn").Visible = True Then Assembly_Shrink
64         Shapes("ExpandBtn").Visible = msoFalse
65     End If
66     End If
67     Application.ScreenUpdating = True
68     End If
69
70 End Sub
```

1	Option Explicit
2	

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("C3" )) Is Nothing Then
6         Range("E4:E6" ).ClearContents 'Clear Customer/Vendor Details
7         Range("E4" ).Validation.Delete 'Delete Validation
8         If Target.Value = "INVOICE" Then
9             Range("E4" ).Validation.Add xlValidateList, , , "=Customer_Name"
10            Else
11                Range("E4" ).Validation.Add xlValidateList, , , "=Vendor_Name"
12            End If
13        End If
14
15        'On Customer/ Vendor Name Change
16        If Not Intersect(Target, Range("E4" )) Is Nothing Then
17            Dim DBRow As Long
18            Dim ShtName As String
19            If Range("C3" ).Value = "INVOICE" Then ShtName = "Customers" Else ShtName =
20                "Vendors"
21            If Range("E4" ).Value <> Empty And Range("B2" ).Value <> Empty Then
22                DBRow = Range("B2" ).Value 'Customer/ Vendor Row
23                Range("E5" ).Value = ThisWorkbook.Sheets(ShtName).Range("B" & DBRow).Value
24                'Address 1
25                Range("E6" ).Value = ThisWorkbook.Sheets(ShtName).Range("C" & DBRow).Value
26                'Address 2
27            End If
28        End If
29
30        'Set footer message
31        If Not Intersect(Target, Range("C7" )) Is Nothing Then Range("D28" ).Value = Range("B8" ).
32            Value
33
34        'On Order Item Name Change (But not on Invoice Load)
35        If Not Intersect(Target, Range("D8:D27" )) Is Nothing And Range("B5" ).Value = False Then
36            Dim FoundItem As Range
37            Dim ItemRow As Long
38            If Target.Value <> Empty Then
39                Set FoundItem = ItemDB.Range("Item_Name" ).Find(Target.Value, , xlValues, xlWhole)
40                If Not FoundItem Is Nothing Then
41                    ItemRow = FoundItem.Row 'Set ItemRow
42                    If Range("C3" ).Value = "INVOICE" Then
43                        Range("E" & Target.Row).Value = ItemDB.Range("H" & ItemRow).Value 'Sales
44                        Description
45                        Range("F" & Target.Row).Value = ItemDB.Range("J" & ItemRow).Value 'Default
46                        Sales Qty
47                        Range("G" & Target.Row).Value = ItemDB.Range("L" & ItemRow).Value 'Default
48                        Sales Price
49                    Else 'Purchase Order
50                        Range("E" & Target.Row).Value = ItemDB.Range("G" & ItemRow).Value 'Purchase
51                        Description
52                        Range("F" & Target.Row).Value = ItemDB.Range("I" & ItemRow).Value 'Default
53                        Purch Qty
54                        Range("G" & Target.Row).Value = ItemDB.Range("K" & ItemRow).Value 'Default
55                        Purch Price /Item Cost
56                    End If
57                End If
58            Else 'Clear Row
59                Range("E" & Target.Row & ":G" & Target.Row).ClearContents
60            End If
61        End If

```

```
52     1 2 End If
53
54     'On Change of Invoice #, Load Invoice
55     If Not Intersect(Target, Range("C5" )) Is Nothing And Range("C5" ).Value <> Empty Then
56         Order_Load 'Load Invoice
57     End Sub
58
59
60
```

1	Option Explicit
2	

1	Option Explicit
2	


```

1 Option Explicit
2 Dim BuildQty As Long, AssembDBRow As Long, ItemRow As Long, PicRow As Long
3 Dim LastItemRow As Long, ItemDBRow As Long, QtyReq As Long, DisQty As Long
4 Dim TopPos As Double, LeftPos As Double, LeftStart As Double
5 Dim PicFolder As String, PicFile As String, AssembText As String
6 Dim ItemShp As Shape
7
8
9 Sub Assembly_Build()
10 With Items
11 If .Range("M39").Value > .Range("M38").Value Then
12 MsgBox "Please make sure the Items Set To Build Quantity is less than or equal to "
13 & .Range("M38").Value
14 Exit Sub
15 End If
16 AssembDBRow = .Range("B4").Value 'Assembly Database Row
17 BuildQty = .Range("M39").Value 'Build Quantiy
18 LastItemRow = .Range("E59").End(xlUp).Row
19 If LastItemRow < 37 Then
20 MsgBox "Please make sure you have added at least 2 items to build your assembly"
21 Exit Sub
22 End If
23 For ItemRow = 36 To LastItemRow
24 ItemDBRow = .Range("S" & ItemRow).Value 'Item Database Row
25 QtyReq = .Range("G" & ItemRow).Value 'Qty Required
26 If .Range("U" & ItemRow).Value <> "Service" Then ItemDB.Range("Q" & ItemDBRow).
27 Value = ItemDB.Range("Q" & ItemDBRow).Value - (QtyReq * BuildQty) 'Deduct Total
28 Build Qty
29 Next ItemRow
30 ItemDB.Range("Q" & AssembDBRow).Value = ItemDB.Range("Q" & AssembDBRow).Value +
31 BuildQty
32 .Range("M39").ClearContents 'Clear Build Qty
33 .Range("J11").Value = .Range("M37").Value 'Update Quantity On Hand
34 MsgBox BuildQty & " " & .Range("F5").Value & "'s assemblies have been built"
35 End With
36 End Sub
37
38
39 Sub Assembly_Disassemble()
40 With Items
41 If .Range("M40").Value > .Range("M37").Value Then
42 MsgBox "Please make sure the Items Set To Disassemble is less than or equal to " &
43 .Range("M37").Value
44 Exit Sub
45 End If
46 AssembDBRow = .Range("B4").Value 'Assembly Database Row
47 DisQty = .Range("M40").Value 'Disassemble Quantiy
48 LastItemRow = .Range("E59").End(xlUp).Row
49 If LastItemRow < 37 Then
50 MsgBox "Please make sure you have added at least 2 items to build your assembly"
51 Exit Sub
52 End If
53 For ItemRow = 36 To LastItemRow
54 ItemDBRow = .Range("S" & ItemRow).Value 'Item Database Row
55 QtyReq = .Range("G" & ItemRow).Value 'Qty Required
56 If .Range("U" & ItemRow).Value <> "Service" Then ItemDB.Range("Q" & ItemDBRow).
57 Value = ItemDB.Range("Q" & ItemDBRow).Value + (QtyReq * DisQty) 'Deduct Total
58 Build Qty
59 Next ItemRow
60 ItemDB.Range("Q" & AssembDBRow).Value = ItemDB.Range("Q" & AssembDBRow).Value -
61 DisQty

```

```

1 2
54 .Range("M40").ClearContents 'Clear Build Qty
55 .Range("J11").Value = .Range("M37").Value 'Update Quantity On Hand
56 MsgBox DisQty & " " & .Range("F5").Value & "'s assemblies have been disassembled"
57 End With
58 End Sub
59
60 Sub Assembly_Expand()
61 With Items
62 'Delete existing Assembly Item Shapes (if any)
63 For Each ItemShp In .Shapes
64 If InStr(ItemShp.Name, "AssembPic") > 0 Then ItemShp.Delete
65 Next ItemShp
66 PicFolder = Admin.Range("C4").Value 'Picture folder
67 LastItemRow = Application.WorksheetFunction.CountA(Items.Range("E36:E58")) + 35
68 If LastItemRow < 37 Then Exit Sub
69 LeftStart = .Range("E17").Left
70 LeftPos = .Range("E17").Left
71 TopPos = .Range("E17").Top
72 PicRow = 1 'Set initial Picture row
73 For ItemRow = 36 To LastItemRow
74 ItemDBRow = .Range("S" & ItemRow).Value 'Item Database Row
75 PicFile = PicFolder & "\" & ItemDB.Range("N" & ItemDBRow).Value 'Picture Path
76 If Dir(PicFile, vbDirectory) = "" Then GoTo NextItem
77 .Pictures.Insert(PicFile).Name = "AssembPic" & ItemRow
78 With .Shapes("AssembPic" & ItemRow)
79 .Left = LeftPos
80 .Top = TopPos
81 .LockAspectRatio = msoCTrue
82 .Width = 60
83 End With
84
85 'Create & Position Connector
86 .Shapes("ConSample").Duplicate.Name = "AssembPicCon" & ItemRow
87 With .Shapes("AssembPicCon" & ItemRow).ConnectorFormat
88 If PicRow = 1 Then 'Top Row
89 .BeginConnect Items.Shapes("ShrinkBtn"), 3
90 Else 'All Other row
91 .BeginConnect Items.Shapes("AssembPicText" & ItemRow - 8), 3
92 End If
93 .EndConnect Items.Shapes("AssembPic" & ItemRow), 1
94 End With
95
96 'Create & Position text box
97 AssembText = .Range("G" & ItemRow).Value & "-" & .Range("E" & ItemRow).Value
98 .Shapes("TextBoxSample").Duplicate.Name = "AssembPicText" & ItemRow
99 With .Shapes("AssembPicText" & ItemRow)
100 .Left = LeftPos
101 .Top = TopPos + 62
102 .TextFrame2.TextRange.Text = AssembText 'Add Qty & Item Name
103 End With
104
105
106 'Update Left & Top Position
107 If LeftPos > .Range("L17").Left Then
108 TopPos = TopPos + 105
109 LeftPos = LeftStart 'Set Back to initial left post
110 PicRow = PicRow + 1
111 Else
112 LeftPos = LeftPos + 90
113 End If
114 NextItem:

```

1 2 3

```
1 2 3
115     Next ItemRow
116         .Shapes("ExpandBtn" ).Visible = msoFalse
117         .Shapes("ShrinkBtn" ).Visible = msoCTrue
118     End With
119 End Sub
120
121
122 Sub Assembly_Shrink()
123     With Items
124         For Each ItemShp In .Shapes
125             If InStr(ItemShp.Name, "AssembPic" ) > 0 Then ItemShp.Delete
126         Next ItemShp
127         .Shapes("ShrinkBtn" ).Visible = msoFalse
128         .Shapes("ExpandBtn" ).Visible = msoCTrue
129     End With
130 End Sub
```

```

1 Option Explicit
2 Dim ItemRow As Long, ItemCol As Long, LastRow As Long, LastResultRow As Long, LastAssembRow
  As Long, AssembDBRow As Long, AssembRow As Long
3 Dim ItemID As String, PicFile As String, ItemPicFolder As String, PicPath As String,
  ResultRow As Long
4 Dim ItemFilePic As FileDialog
5 Dim ItemShp As Shape
6
7 Sub Item_AddItemPic()
8     ItemPicFolder = Admin.Range("C4" ).Value 'Item Picture folder
9     If Dir(ItemPicFolder, vbDirectory) = "" Then
10         MsgBox "Please add a picture file path inside the Admin screen before adding picture"
11         Exit Sub
12     End If
13     With Items
14         Set ItemFilePic = Application.FileDialog(msoFileDialogFilePicker)
15         With ItemFilePic
16             .Title = "Please select an Item picture"
17             .Filters.Add "Picture Files" , "*.jpg,*.png,*.gif" , 1
18             .AllowMultiSelect = False
19             If .Show <> -1 Then GoTo NoSelection
20             PicPath = .SelectedItems(1)
21         End With
22         .Range("L6 " ).Value = Dir(PicPath) 'Set Name (Extracts File Name from file path)
23         On Error Resume Next
24         FileCopy PicPath, ItemPicFolder & "\" & Dir(PicPath) 'Copy picture to app Item
          picture folder
25         On Error GoTo 0
26         Item_ShowPic
27     End With
28     NoSelection:
29 End Sub
30 Sub Item_AddNew()
31     With Items
32         'Delete existing Assembly Item Shapes (if any)
33         For Each ItemShp In .Shapes
34             If InStr(ItemShp.Name, "AssembPic" ) > 0 Then ItemShp.Delete
35         Next ItemShp
36         .Range("ItemData" ).ClearContents 'Clear all of the data those cells
37         .Shapes("ExpandBtn" ).Visible = msoFalse
38         .Shapes("ShrinkBtn" ).Visible = msoFalse
39         .Range("B3" ).ClearContents
40         .Shapes("NewItemGrp" ).Visible = msoCTrue
41         .Shapes("ExistItemGrp" ).Visible = msoFalse
42         Item_ClearItemPic
43     End With
44 End Sub
45
46 Sub Item_Cancel_New()
47     If ItemDB.Range("B4" ).Value <> Empty Then Items.Range("I3" ).Value = ItemDB.Range("B4" ).
  Value 'Set First Item Name To Load
48 End Sub
49 Sub Item_ClearItemPic()
50     On Error Resume Next
51     Items.Shapes("ItemPic" ).Delete
52     On Error GoTo 0
53     Items.Range("L6" ).ClearContents 'Clear picture cell
54 End Sub
55
56 Sub Item_Delete()
57     If MsgBox("Are you sure you want to delete this item?" , vbYesNo, "Delete Item" ) = vbNo

```

```

1
58 Then Exit Sub
59 With Items
60     If .Range("B4").Value = Empty Then
61         MsgBox "Please select a correct Item from the list on the left"
62         Exit Sub
63     End If
64     ItemRow = .Range("B4").Value 'Set Item Row
65     ItemDB.Range(ItemRow & ":" & ItemRow).Delete
66     Item_AddNew
67     MsgBox "Item has been deleted"
68 End With
69 End Sub
70
71 Sub Item_Load()
72 With Items
73     .Range("ItemData").ClearContents 'Clear all of the data those cells
74     If .Range("B4").Value = Empty Then
75         MsgBox "Please select a correct Item from the list on the left"
76         Exit Sub
77     End If
78     'Delete existing Assembly Item Shapes (if any)
79     For Each ItemShp In .Shapes
80         If InStr(ItemShp.Name, "AssembPic") > 0 Then ItemShp.Delete
81     Next ItemShp
82     .Shapes("ExpandBtn").Visible = msoFalse 'Hide Expand button by default
83     .Shapes("ShrinkBtn").Visible = msoFalse 'Hide Expand button by default
84     ItemPicFolder = Admin.Range("C4").Value 'Item Picture folder
85     ItemRow = .Range("B4").Value 'Set Item Row
86     For ItemCol = 2 To 15 'Load Item Data
87         .Range(ItemDB.Cells(1, ItemCol).Value).Value = ItemDB.Cells(ItemRow, ItemCol).Value
88     Next ItemCol
89     If .Range("B1").Value = 4 Then Item_ShowPic 'Add item pic if avail and in Gen. Info
90     tab
91     If .Range("H5").Value = "Assembly" Then 'Load Assembly Items
92         LastAssembRow = AssembDB.Range("A999999").End(xlUp).Row
93         If LastAssembRow < 3 Then GoTo NoItems
94         .Shapes("ExpandBtn").Visible = msoCTrue 'Show Expand Button
95         AssembDB.Range("A2:E" & LastAssembRow).AdvancedFilter xlFilterCopy, CriteriaRange:=
96         AssembDB.Range("G2:G3"), CopyToRange:=AssembDB.Range("I2:L2"), Unique:=True
97         LastResultRow = AssembDB.Range("I99999").End(xlUp).Row
98         If LastResultRow < 3 Then Exit Sub
99         For ResultRow = 3 To LastResultRow
100            ItemRow = AssembDB.Range("K" & ResultRow).Value
101            .Range("E" & ItemRow).Value = AssembDB.Range("I" & ResultRow).Value 'Item Name
102            .Range("G" & ItemRow).Value = AssembDB.Range("J" & ResultRow).Value 'Qty
103            Required
104            .Range("R" & ItemRow).Value = AssembDB.Range("L" & ResultRow).Value 'Database
105            Row
106        Next ResultRow
107        NoItems:
108    End If
109    .Shapes("NewItemGrp").Visible = msoFalse
110    .Shapes("ExistItemGrp").Visible = msoCTrue
111 End With
112 End Sub
113
114 Sub Item_SaveUpdate()
115 With Items
116     If .Range("F5").Value = Empty Then
117         1 2 3

```

```

1 2 3
113 MsgBox "Please make sure to add an Item Name to your Item before saving"
114 Exit Sub
115 End If
116 If .Range("B4" ).Value = Empty Then 'New Item
117 ItemRow = ItemDB.Range("A99999" ).End(xlUp).Row + 1 'First Avail Row.
118 .Range("B3" ).Value = .Range("B5" ).Value 'Item ID
119 ItemDB.Range("A" & ItemRow).Value = .Range("B5" ).Value 'Item ID
120 Else 'Existing Item
121 ItemRow = .Range("B4" ).Value 'Item Row
122 End If
123
124 For ItemCol = 2 To 15 'Not on Qty on Hand formula
125 If ItemCol <> 13 Then ItemDB.Cells(ItemRow, ItemCol).Value = .Range(ItemDB.Cells(1,
ItemCol).Value).Value 'Save Data to DB
126 Next ItemCol
127
128 'Check For Assembly & Items
129 If .Range("H5" ).Value = "Assembly" Then
130 LastAssembRow = .Range("E59" ).End(xlUp).Row 'Last Assembly Item Row
131 If LastAssembRow < 36 Then GoTo NoItems
132 For AssembRow = 36 To LastAssembRow
133 If .Range("R" & AssembRow).Value = "" Then 'New Item
134 AssembDBRow = AssembDB.Range("A99999" ).End(xlUp).Row + 1 ' First Avail Row
135 AssembDB.Range("A" & AssembDBRow).Value = .Range("B3" ).Value 'Item ID
136 AssembDB.Range("D" & AssembDBRow).Value = AssembRow 'Assembly Item row
137 AssembDB.Range("E" & AssembDBRow).Value = "=Row()" 'Assemb DB Row
138 .Range("R" & AssembRow).Value = AssembDBRow
139 Else 'Existing
140 AssembDBRow = .Range("R" & AssembRow).Value 'Existing Assemb. DB Row
141 End If
142 AssembDB.Range("B" & AssembDBRow).Value = .Range("E" & AssembRow).Value 'Item
Name
143 AssembDB.Range("C" & AssembDBRow).Value = .Range("G" & AssembRow).Value 'Item
Qty
144 Next AssembRow
145 End If
146 NoItems:
147 .Shapes("NewItemGrp" ).Visible = msoFalse 'Hide New Item group
148 .Shapes("ExistItemGrp" ).Visible = msoCTrue 'Show existing item grou
149 End With
150 End Sub
151
152 Sub Item_ShowPic()
153 With Items
154 On Error Resume Next
155 .Shapes("ItemPic" ).Delete
156 On Error GoTo 0
157 If .Range("L6" ).Value = Empty Then Exit Sub
158 PicPath = Admin.Range("C4" ).Value & "\" & .Range("L6" ).Value 'Full Picture Path
159 If Dir(PicPath, vbDirectory) = "" Then
160 MsgBox "Picture name or folder is incorrect"
161 Exit Sub
162 End If
163 .Pictures.Insert(PicPath).Name = "ItemPic"
164 With .Shapes("ItemPic" )
165 .LockAspectRatio = msoCTrue
166 If .Width > .Height Then .Width = 90 Else .Height = 90
167 .Left = Items.Range("L7" ).Left + (Items.Range("L:M" ).Width - .Width) / 2
'Centered Left Pos
168 .Top = Items.Range("L7" ).Top + (Items.Range("7:14" ).Height - .Height) / 2
'Centered Top Pos

```

1 2 3

```
169     1 2 3 .Visible = msoCTrue
170     1 2 3 End With
171     1 2 3 End With
172     1 2 3 End Sub
```

```

1  Option Explicit
2
3  Sub Order_CancelNew()
4      If OrderList.Range("A3" ).Value <> Empty Then Orders.Range("C5" ).Value = OrderList.Range(
5          "A3" ).Value
6      End Sub
7
8  Sub Order_Load()
9      Dim OrdRow As Long, OrdItemRow As Long, LastOrdItemRow As Long
10     Dim ResultRow As Long, LastItemResultRow As Long
11     With Orders
12         If .Range("B7" ).Value = Empty Then
13             MsgBox "Please select a correct Order #"
14             Exit Sub
15         End If
16         .Range("H1" ).Value = .Range("C5" ).Value 'Set Order ID
17         OrdRow = .Range("B3" ).Value 'Order Row
18         .Range("C5,E4:E6,H2,D8:G27,I8:I27" ).ClearContents 'Clear Order Fields
19         .Range("B5" ).Value = True 'Set Order Load To True
20         .Range("C3" ).Value = OrderList.Range("B" & OrdRow).Value 'Order Type
21         .Range("H2" ).Value = OrderList.Range("C" & OrdRow).Value 'Date
22         .Range("E4" ).Value = OrderList.Range("D" & OrdRow).Value 'Customer /Vendor
23         .Range("D28" ).Value = OrderList.Range("E" & OrdRow).Value 'Footer Items
24
25         'Load Order Items
26         LastOrdItemRow = OrderItems.Range("A9999" ).End(xlUp).Row 'Last Item Row
27         If LastOrdItemRow < 3 Then GoTo NoItems
28         OrderItems.Range("A2:H" & LastOrdItemRow).AdvancedFilter xlFilterCopy, CriteriaRange:=
29             OrderItems.Range("I2:I3" ), CopyToRange:=OrderItems.Range("K2:R2" ), Unique:=False
30         LastItemResultRow = OrderItems.Range("K9999" ).End(xlUp).Row
31         If LastItemResultRow < 3 Then GoTo NoItems
32         For ResultRow = 3 To LastItemResultRow
33             OrdRow = OrderItems.Range("Q" & ResultRow).Value 'Order Row
34             .Range("D" & OrdRow & ":G" & OrdRow).Value = OrderItems.Range("M" & ResultRow &
35                 ":P" & ResultRow).Value 'Item, Desc, Qty, Price
36             .Range("I" & OrdRow).Value = OrderItems.Range("R" & ResultRow).Value 'Bring over
37                 item row
38         Next ResultRow
39         NoItems:
40         .Range("B4" ).Value = False 'Set new Order to false
41         .Range("B5" ).Value = False 'Set Order Load To False
42         .Shapes("CancelNewBtn" ).Visible = msoFalse
43         .Shapes("AddNewBtn" ).Visible = msoCTrue
44     End With
45 End Sub
46
47 Sub Order_New()
48     With Orders
49         .Range("B4" ).Value = True 'Set new Order to true
50         .Range("H1,E3,E4,E5,D8:G27,I8:I27" ).ClearContents 'Clear Order Fields
51         .Range("H2" ).Value = Date 'Set Current Date
52         .Shapes("CancelNewBtn" ).Visible = msoCTrue
53         .Shapes("AddNewBtn" ).Visible = msoFalse
54     End With
55 End Sub
56
57 Sub Order_Print()
58     Orders.PrintOut , , , False, , , , False
59 End Sub

```



```

58 Sub Order_SaveUpdate()
59 Dim OrdRow As Long, OrdItemRow As Long, LastItemRow As Long, ItemRow As Long
60 Dim OrdType As String
61 With Orders
62     If .Range("E4").Value = Empty Then
63         MsgBox "Please select " & .Range("D4").Value & " from the drop down list before
        saving"
64         Exit Sub
65     End If
66
67     OrdType = .Range("C3").Value 'Set Order Type
68     'Determine New or Existing Order
69     If .Range("B3").Value = Empty Then 'New Order
70         OrdRow = OrderList.Range("A9999").End(xlUp).Row + 1 'First Available Row
71         .Range("H1").Value = .Range("B6").Value 'Add Order Number
72         OrderList.Range("A" & OrdRow).Value = .Range("B6").Value 'Add Order #
73     Else: 'Existing Order
74         OrdRow = .Range("B3").Value 'Existing Order Row
75     End If
76     OrderList.Range("B" & OrdRow).Value = OrdType
77     OrderList.Range("C" & OrdRow).Value = .Range("H2").Value 'Date
78     OrderList.Range("D" & OrdRow).Value = .Range("E4").Value 'Customer /Vendor Name
79     OrderList.Range("E" & OrdRow).Value = .Range("D28").Value 'Footer Message
80     OrderList.Range("F" & OrdRow).Value = .Range("H30").Value 'Order Total
81
82     'Add / Update Order Items
83     LastItemRow = .Range("D27").End(xlUp).Row 'Last Order Item Row
84     If LastItemRow < 8 Then GoTo NoItems
85     For ItemRow = 8 To LastItemRow
86         If .Range("I" & ItemRow).Value <> Empty Then 'existing Row
87             OrdItemRow = .Range("I" & ItemRow).Value
88         Else 'New Row
89             OrdItemRow = OrderItems.Range("A9999").End(xlUp).Row + 1 'First AvailRow
90             OrderItems.Range("A" & OrdItemRow).Value = .Range("H1").Value 'Order #
91             .Range("I" & ItemRow).Value = OrdItemRow 'Order Item Row
92             OrderItems.Range("H" & OrdItemRow).Value = "=Row()" 'Add Row #
93         End If
94         OrderItems.Range("B" & OrdItemRow).Value = OrdType 'Order Type
95         OrderItems.Range("C" & OrdItemRow & ":F" & OrdItemRow).Value = .Range("D" &
        ItemRow & ":G" & ItemRow).Value 'Add Desc, Qty, Price
96         OrderItems.Range("G" & OrdItemRow).Value = ItemRow
97     Next ItemRow
98     NoItems:
99     .Range("B4").Value = False 'Set new Order to false
100     .Shapes("CancelNewBtn").Visible = msoFalse
101     .Shapes("AddNewBtn").Visible = msoCTrue
102 End With
103 End Sub

```

A

Add, 10, 17
 Admin, 15, 17-19
 AdvancedFilter, 18, 21
 AllowMultiSelect, 17
 Application, 6, 7, 15, 17
 AssembDB, 18, 19
 AssembDBRow, 14, 17, 19
 Assembly_Build, 14
 Assembly_Disassemble, 14
 Assembly_Expand, 15
 Assembly_Shrink, 7, 16
 AssembRow, 17, 19
 AssembText, 14, 15

B

BeginConnect, 15
 BuildQty, 14

C

Cells, 18, 19
 ClearContents, 6, 10, 14, 15, 17, 18, 21
 Column, 6
 ConnectorFormat, 15
 CopyToRange, 18, 21
 CountA, 15
 CountLarge, 6, 10
 CriteriaRange, 18, 21

D

DBRow, 10
 Delete, 10, 15-19
 Dir, 15, 17, 19
 DisQty, 14, 15
 Duplicate, 15

E

Empty, 6, 10, 11, 17-19, 21, 22
 EndConnect, 15
 EntireRow, 6, 7
 Explicit, 2-6, 8-10, 12-14, 17, 21

F

FileCopy, 17
 FileDialog, 17
 Filters, 17
 Find, 6, 10
 FoundItem, 6, 10

H

Height, 19
 Hidden, 6, 7

I

Insert, 15, 19
 InStr, 15-18
 Intersect, 6, 10, 11
 Item_AddItemPic, 17
 Item_AddNew, 17, 18
 Item_Cancel_New, 17
 Item_ClearItemPic, 17
 Item_Delete, 17
 Item_Load, 6, 18
 Item_SaveUpdate, 18
 Item_ShowPic, 17-19
 ItemCol, 17-19
 ItemDB, 6, 10, 14, 15, 17-19
 ItemDBRow, 14, 15
 ItemFilePic, 17

ItemID, 17
 ItemNm, 6
 ItemPicFolder, 17, 18
 ItemRow, 6, 10, 14-19, 22
 Items, 14-19
 ItemShp, 14-18

L

LastAssembRow, 17-19
 LastItemResultRow, 21
 LastItemRow, 14, 15, 22
 LastOrdItemRow, 21
 LastResultRow, 17, 18
 LastRow, 17
 Left, 15, 19
 LeftPos, 14, 15
 LeftStart, 14, 15
 LockAspectRatio, 15, 19

M

MsgBox, 6, 14, 15, 17-19, 21, 22
 msoCTrue, 6, 7, 15-22
 msoFalse, 6, 7, 16-19, 21, 22
 msoFileDialogFilePicker, 17

N

Name, 15-19
 NextItem, 15
 NoItems, 18, 19, 21, 22
 NoSelection, 17

O

Order_CancelNew, 21
 Order_Load, 11, 21
 Order_New, 21
 Order_Print, 21
 Order_SaveUpdate, 22
 OrderItems, 21, 22
 OrderList, 21, 22
 Orders, 21, 22
 OrdItemRow, 21, 22
 OrdRow, 21, 22
 OrdType, 22

P

PicFile, 14, 15, 17
 PicFolder, 14, 15
 PicPath, 17, 19
 PicRow, 14, 15
 Pictures, 15, 19
 PrintOut, 21

Q

QtyReq, 14

R

Range, 6, 7, 10, 11, 14, 15, 17-19, 21, 22
 ResultRow, 17, 18, 21
 Row, 6, 10, 14, 18, 19, 21, 22

S

ScreenUpdating, 6, 7
 SelectedItems, 17
 Shape, 14, 17
 Shapes, 6, 7, 15-19, 21, 22
 Sheets, 10
 Show, 17
 ShtName, 10

T

Target, 6, 10, 11
 Text, 15
 TextFrame2, 15
 TextRange, 15
 ThisWorkbook, 10
 Title, 17
 Top, 15, 19
 TopPos, 14, 15

U

Unique, 18, 21

V

Validation, 10
 Value, 6, 7, 10, 11, 14, 15, 17-19, 21, 22
 vbDirectory, 15, 17, 19
 vbNo, 17
 vbYesNo, 17
 Visible, 6, 7, 16-22

W

Width, 15, 19
 Worksheet_Change, 6, 10
 Worksheet_SelectionChange, 6
 WorksheetFunction, 15

X

xlFilterCopy, 18, 21
 xlUp, 14, 18, 19, 21, 22
 xlValidateList, 10
 xlValues, 6, 10
 xlWhole, 6, 10

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