

VBA SOURCE CODE BOOK



PERISHABLE INVENTORY MANAGEMENT

FREE
TEMPLATE
DOWNLOAD!

How To Manage Perishable
Inventory With Sell-By Dates
in Excel



[DOWNLOAD
APPLICATION](#)



[VIEW
TRAINING](#)

by: *Randy Austin*

ABOUT THE AUTHOR

A 4-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 471,000 YouTube subscribers, 35,448,742 video views, 430+ comprehensive training videos, and a thriving 65,000 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](#)

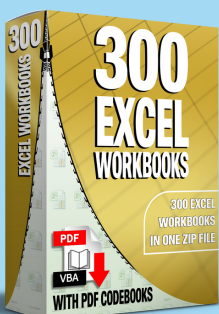
-



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.

[Click here to learn more](#)

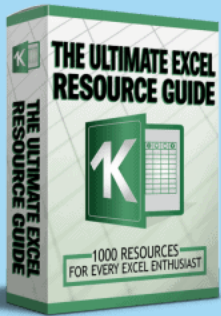
-



Incredible Package of 300 of my BEST Applications with PDF VBA Codebooks packed into a SINGLE ZIP File which also includes the "300 Workbook Library".

[Click here to learn more](#)

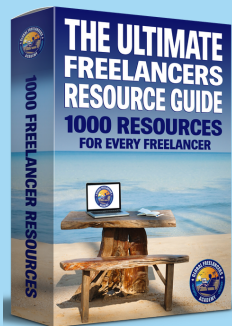
OUR COURSES & PRODUCTS



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](#)

-



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](#)

-



Revolutionize the way you work with Excel and take productivity to the next level with the Excel AI Toolpack - the FIRST AI tool designed for ANY Windows Desktop version of Excel. This incredible add-in combines FIVE powerful AI tools, transforming your Excel into an intelligent powerhouse!

[Click here to learn more](#)

Table of Contents

Projects.....	3
VBAProject.....	3
Documents.....	3
Admin.....	3
(Declarations).....	3
Worksheet_SelectionChange [Sub].....	3
Dashboard.....	4
(Declarations).....	4
ExplInv.....	5
(Declarations).....	5
Worksheet_Change [Sub].....	5
Worksheet_SelectionChange [Sub].....	5
Orders.....	6
(Declarations).....	6
Products.....	7
(Declarations).....	7
Purchases.....	8
(Declarations).....	8
Worksheet_Change [Sub].....	8
Worksheet_SelectionChange [Sub].....	8
PurchItems.....	9
(Declarations).....	9
Sales.....	10
(Declarations).....	10
Worksheet_Change [Sub].....	10
Worksheet_SelectionChange [Sub].....	10
SalesItems.....	11
(Declarations).....	11
Suppliers.....	12
(Declarations).....	12
ThisWorkbook.....	13
(Declarations).....	13
Modules.....	14
Admin_Macros.....	14
(Declarations).....	14
Admin_AddIcon [Sub].....	14
Admin_ClearIcon [Sub].....	14
BrowseForPicFolder [Sub].....	14
MainMenu [Sub].....	14
Dashboard_Macros.....	16
(Declarations).....	16
Dashboard_Refresh [Sub].....	16
Inventory_Mgmt.....	17
(Declarations).....	17
ExplInv_GoToPO [Sub].....	17
ExplInv_Refresh [Sub].....	17
ExplInv_RemoveItems [Sub].....	17
PurchOrder_Macros.....	19
(Declarations).....	19
PurchOrder_AddNew [Sub].....	19
PurchOrder_Delete [Sub].....	19
PurchOrder_ListLoad [Sub].....	19
PurchOrder_Load [Sub].....	20
PurchOrder_Print [Sub].....	21
PurchOrder_SaveUpdate [Sub].....	21
SalesOrder_Macros.....	23
(Declarations).....	23
SalesOrder_AddNew [Sub].....	23
SalesOrder_Delete [Sub].....	23
SalesOrder_ListLoad [Sub].....	23
SalesOrder_Load [Sub].....	24
SalesOrder_Print [Sub].....	25
SalesOrder_SaveUpdate [Sub].....	25
SalesOrder_ScreenMacros.....	28
(Declarations).....	28
FoodTypes_Refresh [Sub].....	28

Table of Contents

LoadProducts [Sub]	28
Prod_Select [Sub]	29
TypeSelect [Sub]	30
UpdateOrderTotals [Sub]	30

```
1  Option Explicit
2
3
4  Private Sub Worksheet_SelectionChange(ByVal Target As Range)
5      If Target.CountLarge > 1 Then Exit Sub
6      If Shapes("IconGrp" ).Visible = True Then Shapes("IconGrp" ).Visible = msoFalse
7      'On selection of any resource row
8      If Not Intersect(Target, Range("B6:C16" )) Is Nothing And Range("B" & Target.Row).
          Value <> Empty Then
9          Range("AA1" ).Value = Target.Row 'Set Selected Row
10         With Shapes("Icongrp" )
11             .Left = Range("D" & Target.Row).Left
12             .Top = Range("D" & Target.Row).Top
13             .Visible = msoCTrue
14         End With
15     End If
16
17
18 End Sub
```

1 Option Explicit

2


```

1  Option Explicit
2
3  Private Sub Worksheet_Change(ByVal Target As Range)
4      'On Sell By Date Change, Refresh list
5      If Not Intersect(Target, Range("P4" )) Is Nothing And Range("P4" ).Value <> Empty
6          Then ExpInv_Refresh
7      End Sub
8
9  Private Sub Worksheet_SelectionChange(ByVal Target As Range)
10     If Target.CountLarge > 1 Then Exit Sub
11     'On Selection of any row in the table set selected row
12     If Not Intersect(Target, Range("F6:P999" )) Is Nothing And Range("F" & Target.Row).
13         Value <> Empty Then Range("B5" ).Value = Target.Row
14
15     'On Selection of Check All/None
16     If Not Intersect(Target, Range("E5" )) Is Nothing Then
17         Dim LastRow As Long
18         LastRow = Range("E99999" ).End(xlUp).Row
19         If Target.Value = Chr(111) Then 'Current Unchecked
20             Range("E5:E" & LastRow).Value = Chr(254) 'Check
21             Range("P6:P" & LastRow).Value = Range("O6:O" & LastRow).Value 'Add All Qty
22             In Stock to Remove Qty
23         Else 'Checked
24             Range("E5:E" & LastRow).Value = Chr(111) 'Uncheck
25             Range("P6:P" & LastRow).ClearContents
26         End If
27         Range("P4" ).Select 'Select Other Cell
28     End If
29
30     'On Selection of a single checkbox
31     If Not Intersect(Target, Range("E6:E999" )) Is Nothing And Range("E" & Target.Row)
32     <> "" Then
33         If Target.Value = Chr(111) Then
34             Target.Value = Chr(254)
35             Range("P" & Target.Row).Value = Range("O" & Target.Row).Value
36         Else
37             Target.Value = Chr(111)
38             Range("P" & Target.Row).ClearContents
39         End If
40         Range("E4" ).Select 'Select Other Cell
41     End If
42 End Sub

```

1 Option Explicit

2

1 Option Explicit

2

```
1 Option Explicit
```

```
2
3
4 Private Sub Worksheet_Change(ByVal Target As Range)
5     If Target.CountLarge > 1 Then Exit Sub
6     If Not Intersect(Target, Range("O8" )) Is Nothing And Range("O8" ).Value <> Empty
7         Then
8             If Range("B3" ).Value = Empty Then
9                 MsgBox "Please enter a correct Purchase Order #"
10                Range("O8" ).ClearContents
11                Exit Sub
12            End If
13            Range("J2" ).Value = Range("O8" ).Value 'Set PO #
14            Range("O8" ).ClearContents
15            PurchOrder_Load 'Run Macro to load Purchase order
16        End If
17
18        If Not Intersect(Target, Range("E8:E9999" )) Is Nothing And Range("E" & Target.Row)
19            .Value <> "" And Range("B1" ).Value = False Then
20                Dim ProdRow As Long
21                Dim ProdID As String
22                On Error Resume Next
23                ProdRow = Products.Range("Prod_Name" ).Find(Target.Value, , xlValues, xlWhole).
24                Row
25                On Error GoTo 0
26                If ProdRow <> 0 Then
27                    Range("D" & Target.Row).Value = Products.Range("A" & ProdRow).Value
28                    'Product ID
29                    Range("F" & Target.Row).Value = Products.Range("C" & ProdRow).Value
30                    'UPC/SKU
31                    Range("G" & Target.Row).Value = Date + Products.Range("F" & ProdRow).Value
32                    'Sell By Date
33                    Range("H" & Target.Row).Value = Products.Range("E" & ProdRow).Value 'Unit
34                    Range("I" & Target.Row).Value = 1 'Default Qty
35                    Range("J" & Target.Row).Value = Products.Range("G" & ProdRow).Value
36                    'Purchase Price
37                Else
38                    MsgBox "Please select a correct Product from the drop down list"
39                    Exit Sub
40                End If
41            End If
42        End Sub
```

```
36
37 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
38     If Target.CountLarge > 1 Then Exit Sub
39     If Not Intersect(Target, Range("M11:O999" )) Is Nothing And Range("M" & Target.Row)
40         .Value <> Empty Then
41             Range("J2" ).Value = Range("M" & Target.Row).Value 'Set PO ID
42             Range("B2" ).Value = Target.Row
43             PurchOrder_Load 'Run Macro to load PO
44         End If
45     End Sub
```

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("U4" )) Is Nothing And Range("U4" ).Value <> Empty
6         Then
7             If Range("B3" ).Value = Empty Then
8                 MsgBox "Please enter a correct Sales Order #"
9                 Range("U4" ).ClearContents
10                Exit Sub
11            End If
12            Range("O5" ).Value = Range("U4" ).Value 'Set Sales Order
13            Range("U4" ).ClearContents
14            SalesOrder_Load 'Run Macro to load Sales Order
15        End If
16
17        If Not Intersect(Target, Range("K8:K9999" )) Is Nothing And Range("K" & Target.Row)
18            .Value <> "" And Range("B1" ).Value = False Then
19                Dim ProdRow As Long
20                Dim ProdID As String
21                On Error Resume Next
22                ProdRow = Products.Range("Prod_Name" ).Find(Target.Value, , xlValues, xlWhole).
23                Row
24                On Error GoTo 0
25                If ProdRow <> 0 Then
26                    Range("J" & Target.Row).Value = Products.Range("A" & ProdRow).Value
27                    'Product ID
28                    Range("L" & Target.Row).Value = Products.Range("C" & ProdRow).Value
29                    'UPC/SKU
30                    Range("N" & Target.Row).Value = Products.Range("E" & ProdRow).Value 'Unit
31                    Range("M" & Target.Row).Value = 1 'Default Qty
32                    Range("O" & Target.Row).Value = Products.Range("H" & ProdRow).Value 'Sales
33                    Price
34                    Range("P" & Target.Row).Value = "=M" & Target.Row & "*O" & Target.Row 'Set
35                    Total Formula
36                Else
37                    MsgBox "Please select a correct Product from the drop down list"
38                    Exit Sub
39                End If
40            End If
41        End If
42    End Sub
43
44 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
45     If Target.CountLarge > 1 Then Exit Sub
46     If Not Intersect(Target, Range("S7:U999" )) Is Nothing And Range("S" & Target.Row).
47         Value <> Empty Then
48         Range("O5" ).Value = Range("S" & Target.Row).Value 'Set PO ID
49         Range("B6" ).Value = Target.Row
50         SalesOrder_Load 'Run Macro to load Sales Order
51     End If
52 End Sub

```

1 Option Explicit

2

1	Option Explicit
2	

1 Option Explicit

2

```

1 Option Explicit
2 Dim SelRow As Long, IconName As String, PicPath As String
3
4
5
6 Sub Admin_AddIcon()
7     Dim IconPic As FileDialog
8     With Admin
9         If .Range("AA1").Value = "" Then Exit Sub
10        SelRow = .Range("AA1").Value
11        IconName = "Icon" & SelRow 'Set Item Name
12        On Error Resume Next
13        .Shapes(IconName).Delete 'Delete any existing icon with the same name
14        On Error GoTo 0
15
16        Set IconPic = Application.FileDialog(msoFileDialogFilePicker)
17        With IconPic
18            .Title = "Select Icon"
19            .Filters.Add "Select Icon Picture", "*.jpg, *.png, *.jpeg, *.gif", 1
20            If .Show <> -1 Then GoTo NoSelection
21            PicPath = .SelectedItems(1)
22        End With
23        .Pictures.Insert(PicPath).Name = IconName 'Add & Name Icon
24        With .Shapes(IconName)
25            .LockAspectRatio = msoCTrue
26            If .Width > .Height Then .Width = 17 Else .Height = 15
27            .Left = Admin.Range("C" & SelRow).Left + (Admin.Range("C:C").Width - .Width) / 2
28            .Top = Admin.Range("C" & SelRow).Top + (Admin.Range(SelRow & ":" & SelRow).Height - .Height) / 2
29        End With
30        NoSelection:
31    End With
32 End Sub
33
34 Sub Admin_ClearIcon()
35     With Admin
36         If .Range("AA1").Value = "" Then Exit Sub
37         SelRow = .Range("AA1").Value
38         IconName = "Icon" & SelRow 'Set Item Name
39         On Error Resume Next
40         .Shapes(IconName).Delete 'Delete any existing icon with the same name
41         On Error GoTo 0
42         .Range("AA1").ClearContents
43     End With
44 End Sub
45
46
47 Sub BrowseForPicFolder()
48     Dim PicFolder As FileDialog
49     Set PicFolder = Application.FileDialog(msoFileDialogFolderPicker)
50     With PicFolder
51         .Title = "Browse Picture Folder"
52         .AllowMultiSelect = False
53         If .Show <> -1 Then GoTo NoSelection
54         Admin.Range("C3").Value = .SelectedItems(1)
55     End With
56     NoSelection:
57 End Sub
58 Sub MainMenu()

```

1

```
1  
59 Dim ShtName As String  
60 ActiveSheet.Shapes("Menu" ).Top = 20 'Reset Top Position to ensure uniformity  
61 ShtName = ActiveSheet.Shapes(Application.Caller).TextFrame2.TextRange.Text  
62 Sheets(ShtName).Activate  
63 End Sub
```

```
1 Option Explicit
2 Dim ProdPic As Shape
3 Dim LeftPos As Double, TopPos As Double, ProdRow As Long
4 Dim PicName As String, PicFolder As String, PicPath As String
5 Sub Dashboard_Refresh()
6     ExpInv.Range("P4" ).Value = Date + 5
7     ExpInv_Refresh 'Run Macro to update expired items
8     With Dashboard
9         For Each ProdPic In .Shapes 'Clear Any Product Pictures
10            If InStr(ProdPic.Name, "ProdPic" ) > 0 Then ProdPic.Delete
11        Next ProdPic
12        PicFolder = [PictureFolder]
13        If PicFolder = Empty Then GoTo NoPicFolder
14        LeftPos = .Range("R1" ).Left 'Left Postion
15        TopPos = .Range("R4" ).Top 'Intial Top Position
16        For ProdRow = 4 To 13
17            PicName = .Range("AJ" & ProdRow).Value
18            PicPath = PicFolder & "\" & PicName 'Picture Path
19            If PicName <> "" Then
20                If Dir(PicPath, vbDirectory) = "" Then GoTo NextProd
21                .Pictures.Insert(PicPath).Name = "ProdPic" & ProdRow
22                With .Shapes("ProdPic" & ProdRow)
23                    .LockAspectRatio = msoCTrue
24                    If .Width > .Height Then .Width = 40 Else .Height = 40
25                    .Left = LeftPos
26                    .Top = TopPos
27                    TopPos = TopPos + 52
28                End With
29            End If
30        Next ProdRow
31        NextProd:
32        NoPicFolder:
33    End With
34 End Sub
```

```

1 Option Explicit
2 Dim LastRow As Long, LastResultRow As Long, ItemRow As Long, PONumb As Long
3 Dim PODBRow As Long, SelRow As Long, RemainQty As Long, RemoveQty As Long
4
5
6 Sub ExpInv_GoToPO()
7     With ExpInv
8         If .Range("B5").Value = Empty Then
9             MsgBox "Please select on a Purchase Order row to view"
10            Exit Sub
11        End If
12        SelRow = .Range("B5").Value 'Selected Row
13        PONumb = Range("F" & SelRow).Value
14        Purchases.Range("O8").Value = PONumb
15        Purchases.Activate
16    End With
17 End Sub
18 Sub ExpInv_Refresh()
19     ExpInv.Range("B5,D6:P9999").ClearContents 'Clear Existing Records
20     With PurchItems
21         LastRow = .Range("A9999").End(xlUp).Row
22         If LastRow < 4 Then Exit Sub
23         .Range("A3:P" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
24             "AR2:AS3"), CopyToRange:=.Range("AV2:BG2"), Unique:=True
25         LastResultRow = .Range("AV9999").End(xlUp).Row
26         If LastResultRow < 3 Then
27             MsgBox "There are no products that meet this filter"
28             Exit Sub
29         End If
30         .Range("AW3:AW" & LastResultRow).Value = Chr(111) 'Add Empty Checkboxes
31         'Sort Based On earliest Expiration date
32         With .Sort
33             .SortFields.Clear
34             .SortFields.Add Key:=PurchItems.Range("BC3"), SortOn:=xlSortOnValues, Order:=
35                 xlAscending, DataOption:=xlSortNormal 'Sort
36             .SetRange PurchItems.Range("AV3:BG" & LastResultRow) 'Set Range
37             .Apply 'Apply Sort
38         End With
39         ExpInv.Range("D6:O" & LastResultRow + 3).Value = .Range("AV3:BG" &
40             LastResultRow).Value 'Bring Over Products
41     End With
42 End Sub
43 Sub ExpInv_RemoveItems()
44     Dim InvCount As Long
45     With ExpInv
46         LastRow = .Range("P99999").End(xlUp).Row
47         If LastRow < 6 Then
48             MsgBox "Please update at least one Qty within the Remove Qty Column"
49             Exit Sub
50         End If
51         'Check All Quantities before making any changes
52         For ItemRow = 6 To LastRow
53             If .Range("P" & ItemRow).Value <> "" And .Range("P" & ItemRow).Value > .
54                 Range("O" & ItemRow).Value Then
55                 MsgBox "Please make sure the Remove Qty for " & .Range("H" & ItemRow).
56                     Value & " is less than or equal to the Qty In Stock"
57                 Exit Sub
58             End If
59         Next ItemRow
60     End With
61 End Sub

```

1 2 3

```
1 2 3
56 If IsNumeric(.Range("P" & ItemRow).Value) = False Then
57     MsgBox "Please make sure the 'Remove Qty' is a number for " & .Range("H"
    & ItemRow).Value
58     Exit Sub
59 End If
60 Next ItemRow
61
62 'Run Loop again to make updates
63 For ItemRow = 6 To LastRow
64     If .Range("P" & ItemRow).Value <> Empty Then
65         PODBRow = .Range("D" & ItemRow).Value
66         PurchItems.Range("J" & PODBRow).Value = PurchItems.Range("J" & PODBRow).
            Value + Range("P" & ItemRow).Value 'Increase Qty Unsold/Removed
67         InvCount = InvCount + 1 'Increment count
68     End If
69 Next ItemRow
70 MsgBox InvCount & " products have been updated with new unsold quantities"
71 ExpInv_Refresh
72 End With
73 End Sub
```

```

1 Option Explicit
2 Dim OrderRow As Long, ItemRow As Long, LastRow As Long, LastItemRow As Long, ItemDBRow
  As Long, LastResultRow As Long, ResultRow As Long
3 Dim OrderType As String, NameType As String
4 Dim ProdID As Long, OrderID As String
5 Sub PurchOrder_AddNew()
6     With Purchases
7         .Range("B2,J2:K2,F4,D8:J99999,L8:L99999").ClearContents
8         .Range("J2").Value = .Range("B5").Value 'Next Order ID
9         .Range("J3").Value = Now 'Current Date/Time
10        .Range("F4").Select 'Select Vendor
11    End With
12 End Sub
13
14 Sub PurchOrder_Delete()
15     With Purchases
16         If MsgBox("Are you sure you want to delete this Purchase Order?" , vbYesNo,
  "Delete Purchase Order" ) = vbNo Then Exit Sub
17         If .Range("B4").Value = Empty Then GoTo NotSaved
18         OrderRow = .Range("B4").Value
19         Orders.Range(OrderRow & ":" & OrderRow).EntireRow.Delete
20
21         'Delete Order Items
22         With PurchItems
23             LastRow = .Range("A99999").End(xlUp).Row
24             If LastRow < 4 Then Exit Sub
25             .Range("A3:O" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
  "AE2:AE3" ), CopyToRange:=.Range("AG2:A02" ), Unique:=True
26             LastResultRow = .Range("AG99999").End(xlUp).Row
27             If LastResultRow < 3 Then Exit Sub
28             For ResultRow = 3 To LastResultRow
29                 ItemDBRow = .Range("AO" & ResultRow).Value 'Item DB Row
30                 .Range("A" & ItemDBRow & ":O" & ItemDBRow).ClearContents 'Clear Contents
31             Next ResultRow
32
33             'Sort remaining fields to clear blank rows (sort By Order ID)
34             With .Sort
35                 .SortFields.Clear
36                 .SortFields.Add Key:=PurchItems.Range("A4" ), SortOn:=xlSortOnValues, Order
  :=xlAscending, DataOption:=xlSortNormal 'Sort
37                 .SetRange PurchItems.Range("A4:O" & LastRow) 'Set Range
38                 .Apply 'Apply Sort
39             End With
40         End With
41         NotSaved:
42         PurchOrder_ListLoad 'Reload List
43         PurchOrder_AddNew
44     End With
45 End Sub
46
47 Sub PurchOrder_ListLoad()
48     Purchases.Range("M11:O999").ClearContents 'Clear Previous list
49     With Orders
50         LastRow = .Range("A99999").End(xlUp).Row
51         If LastRow < 4 Then Exit Sub
52         .Range("H3").Value = "Purchase" 'Set Criteria
53         .Range("A3:E" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
  "H2:H3" ), CopyToRange:=.Range("K2:M2" ), Unique:=True
54         LastResultRow = .Range("K99999").End(xlUp).Row

```

1 2

```

1 2
55 If LastResultRow < 3 Then Exit Sub
56 If LastResultRow < 4 Then GoTo SkipSort
57 'Sort Most recent first
58 With .Sort
59     .SortFields.Clear
60     .SortFields.Add Key:=Orders.Range("M3" ), SortOn:=xlSortOnValues, Order:=
xlDescending, DataOption:=xlSortNormal 'Sort
61     .SetRange Orders.Range("K3:M" & LastResultRow) 'Set Range
62     .Apply 'Apply Sort
63 End With
64
65 SkipSort:
66 Purchases.Range("M11:O" & LastResultRow + 8).Value = .Range("K3:M" &
LastResultRow).Value
67 If Purchases.Range("J2" ).Value <> Empty Then
68     OrderID = Purchases.Range("J2" ).Value
69     On Error Resume Next
70     Purchases.Range("B2" ).Value = Purchases.Range("M11:M999" ).Find(OrderID, ,
xlValues, xlWhole).Row 'Set Selected Order Row
71     On Error GoTo 0
72 End If
73 End With
74 End Sub
75
76 Sub PurchOrder_Load()
77 With Purchases
78     .Range("F4,D8:J99999,L8:L99999" ).ClearContents
79     If .Range("B4" ).Value = Empty Then
80         MsgBox "Please enter a correct Order # to load"
81         Exit Sub
82     End If
83     OrderRow = .Range("B4" ).Value
84     OrderType = Orders.Range("C" & OrderRow).Value
85     If OrderType = "Sales" Then
86         MsgBox "The order # Entered is a Sales Order. Please enter Purhchase order #
again"
87         Exit Sub
88     End If
89     Application.ScreenUpdating = False
90     .Range("J3" ).Value = Orders.Range("B" & OrderRow).Value 'Order Date & Time
91     .Range("F4" ).Value = Orders.Range("D" & OrderRow).Value 'Vendor Name
92 End With
93 'Load Items
94 With PurchItems
95     Purchases.Range("B1" ).Value = True 'Set Order Load To True
96     LastRow = .Range("A99999" ).End(xlUp).Row
97     If LastRow < 4 Then Exit Sub
98     .Range("A3:O" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
"AE2:AE3" ), CopyToRange:=.Range("AG2:A02" ), Unique:=True
99     LastResultRow = .Range("AG99999" ).End(xlUp).Row
100    If LastResultRow < 3 Then Exit Sub
101    For ResultRow = 3 To LastResultRow
102        ItemRow = .Range("AN" & ResultRow).Value 'Item Row
103        Purchases.Range("D" & ItemRow & ":J" & ItemRow).Value = .Range("AG" &
ResultRow & ":AM" & ResultRow).Value 'Item details
104        Purchases.Range("L" & ItemRow & ":L" & ItemRow).Value = .Range("A0" &
ResultRow & ":A0" & ResultRow).Value 'DB Row
105    Next ResultRow
106    Purchases.Range("B1" ).Value = False 'Set Order Load To False
107 End With

```



```

1
108 Application.ScreenUpdating = True
109 End Sub
110
111 Sub PurchOrder_Print()
112 With Purchases
113 LastRow = .Range("E99999").End(xlUp).Row
114 .PageSetup.PrintArea = "$E$1:$K$" & LastRow
115 .PrintOut , , , True, , , False
116 End With
117 End Sub
118
119 Sub PurchOrder_SaveUpdate()
120 With Purchases
121 If .Range("F4").Value = Empty Then
122 MsgBox "Please make sure to enter a Vendor for this purchase order before
saving"
123 .Range("F4").Select
124 Exit Sub
125 End If
126 LastItemRow = .Range("D99999").End(xlUp).Row
127 If LastItemRow < 8 Then
128 MsgBox "Please make sure to enter at least one product item to save this
Purchase Order"
129 Exit Sub
130 End If
131 If .Range("B4").Value = Empty Then 'New Order
132 OrderRow = Orders.Range("A99999").End(xlUp).Row + 1
133 Orders.Range("A" & OrderRow).Value = .Range("J2").Value 'Order #
134 Else 'Existing Order
135 OrderRow = .Range("B4").Value 'Existing Order DB Row
136 End If
137 Orders.Range("B" & OrderRow).Value = .Range("J3").Value 'Order Date & Time
138 Orders.Range("C" & OrderRow).Value = "Purchase" 'Set Order Type
139 Orders.Range("D" & OrderRow).Value = .Range("F4").Value 'Vendor Name
140 Orders.Range("E" & OrderRow).Value = .Range("J6").Value 'Order Total
141
142 'Save/Udpate Items
143 For ItemRow = 8 To LastItemRow
144 If .Range("L" & ItemRow).Value = Empty Then 'New Item DB
145 ItemDBRow = PurchItems.Range("A99999").End(xlUp).Row + 1 'First Avail
Row
146 PurchItems.Range("A" & ItemDBRow).Value = .Range("J2").Value 'Order #
147 PurchItems.Range("K" & ItemDBRow).Value = "=" & ItemDBRow & "-" &
"Sum(I" & ItemDBRow & ":J" & ItemDBRow & ")" 'Add Remain. Qty Formula
148 PurchItems.Range("N" & ItemDBRow).Value = ItemRow 'Set Item Row
149 PurchItems.Range("O" & ItemDBRow).Value = "=Row()"
150 .Range("L" & ItemRow).Value = ItemDBRow
151 Else 'Existing
152 ItemDBRow = .Range("L" & ItemRow).Value 'Existing Order Items DB Row
153 End If
154 ProdID = .Range("D" & ItemRow).Value 'Product ID
155 PurchItems.Range("B" & ItemDBRow).Value = .Range("J3").Value 'Order Date &
Time
156 PurchItems.Range("C" & ItemDBRow).Value = ProdID 'Product ID
157 PurchItems.Range("D" & ItemDBRow).Value = .Range("F" & ItemRow).Value
'UPC/SKU
158 PurchItems.Range("E" & ItemDBRow).Value = .Range("E" & ItemRow).Value
'Product Name
159 PurchItems.Range("F" & ItemDBRow).Value = .Range("G" & ItemRow).Value 'Sell
By Date

```

1 2 3

```
160      1 2 3 PurchItems.Range("G" & ItemDBRow).Value = .Range("H" & ItemRow).Value '
Unit
161      PurchItems.Range("H" & ItemDBRow).Value = .Range("I" & ItemRow).Value
'Batch Purch Qty
162      PurchItems.Range("L" & ItemDBRow).Value = .Range("J" & ItemRow).Value 'Item
Cost
163      PurchItems.Range("M" & ItemDBRow).Value = .Range("K" & ItemRow).Value
'Total Cost
164      Next ItemRow
165      NoItems:
166      PurchOrder_ListLoad 'Reload List
167      MsgBox OrderType & "Purchased Order Saved"
168      End With
169      End Sub
```

```

1 Option Explicit
2 Dim OrderRow As Long, ItemRow As Long, LastRow As Long, LastItemRow As Long
3 Dim PurchItemDBRow As Long, SalesItemDBRow As Long
4 Dim LastPurchItemDBRow As Long, LastResultRow As Long, ResultRow As Long
5 Dim PurchQty As Long, TotalRemainQty As Long, RemainQty As Long
6 Dim OrderType As String, NameType As String, OrderID As String
7
8 Sub SalesOrder_AddNew()
9     With Sales
10        .Range("B7,B10,J9:R99999" ).ClearContents
11        .Range("O5" ).Value = .Range("B4" ).Value 'Next Order ID
12        .Range("O7" ).Value = Now 'Set Current Date & time
13    End With
14 End Sub
15
16 Sub SalesOrder_Delete()
17     With Sales
18        If MsgBox("Are you sure you want to delete this Sales Order?" , vbYesNo, "Delete
19        Sales Order" ) = vbNo Then Exit Sub
20        If .Range("B2" ).Value = Empty Then GoTo NotSaved
21        OrderRow = .Range("B2" ).Value
22        Orders.Range(OrderRow & ":" & OrderRow).EntireRow.Delete
23
24        LastItemRow = .Range("K99999" ).End(xlUp).Row
25        If LastItemRow < 9 Then GoTo NoItems
26        For ItemRow = 9 To LastItemRow
27            'Reduced Sold Items From Purchase Order DB
28            If .Range("R" & ItemRow).Value <> Empty Then
29                PurchItemDBRow = .Range("R" & ItemRow).Value 'Purchase Item DB Row
30                PurchItems.Range("I" & PurchItemDBRow).Value = PurchItems.Range("I" &
31                PurchItemDBRow).Value - .Range("M" & ItemRow).Value 'Reduce Qty Sold
32                If PurchItems.Range("I" & PurchItemDBRow).Value < 0 Then PurchItems.Range(
33                "I" & PurchItemDBRow).Value = 0 'Check For less than 0
34            End If
35            'Clear Out Sales item DB
36            If .Range("Q" & ItemRow).Value <> Empty Then
37                SalesItemDBRow = .Range("Q" & ItemRow).Value 'Sales Item DB Row
38                SalesItems.Range("A" & SalesItemDBRow & ":L" & SalesItemDBRow).
39                ClearContents 'Clear Product Item Details
40            End If
41        Next ItemRow
42    End With
43    'Sort remaining fields to clear blank rows (sort By Order ID)
44    LastRow = SalesItems.Range("A99999" ).End(xlUp).Row
45    With SalesItems.Sort
46        .SortFields.Clear
47        .SortFields.Add Key:=SalesItems.Range("A4" ), SortOn:=xlSortOnValues, Order:=
48        xlAscending, DataOption:=xlSortNormal 'Sort
49        .SetRange SalesItems.Range("A4:L" & LastRow) 'Set Range
50        .Apply 'Apply Sort
51    End With
52    NoItems:
53    NotSaved:
54    SalesOrder_AddNew
55    SalesOrder_ListLoad
56 End Sub
57
58 Sub SalesOrder_ListLoad()
59     Sales.Range("B6,S7:U999" ).ClearContents 'Clear Previous list
60
61

```

1

```

1
56 With Orders
57 LastRow = .Range("A99999").End(xlUp).Row
58 If LastRow < 4 Then Exit Sub
59 .Range("H3").Value = "Sales" 'Set Criteria
60 .Range("A3:E" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
    "H2:H3"), CopyToRange:=.Range("P2:R2"), Unique:=True
61 LastResultRow = .Range("P99999").End(xlUp).Row
62 If LastResultRow < 3 Then Exit Sub
63 If LastResultRow < 4 Then GoTo SkipSort
64
65 'Sort Most recent first
66 With .Sort
67 .SortFields.Clear
68 .SortFields.Add Key:=Orders.Range("Q3"), SortOn:=xlSortOnValues, Order:=
    xlDescending, DataOption:=xlSortNormal 'Sort
69 .SetRange Orders.Range("P3:R" & LastResultRow) 'Set Range
70 .Apply 'Apply Sort
71 End With
72
73 SkipSort:
74 Sales.Range("S7:U" & LastResultRow + 4).Value = .Range("P3:R" & LastResultRow).
    Value
75 End With
76 'Add Selected Row
77 If Sales.Range("O5").Value <> Empty Then
78 OrderID = Sales.Range("O5").Value
79 On Error Resume Next
80 Sales.Range("B6").Value = Sales.Range("S7:S999").Find(OrderID, , xlValues,
    xlWhole).Row 'Set Selected Order Row
81 On Error GoTo 0
82 End If
83 End Sub
84
85 Sub SalesOrder_Load()
86 With Sales
87 .Range("B7,J9:R99999").ClearContents
88 If .Range("B2").Value = Empty Then
89 MsgBox "Please enter a correct Order # to load"
90 Exit Sub
91 End If
92 OrderRow = .Range("B2").Value
93 OrderType = Orders.Range("C" & OrderRow).Value
94 If OrderType = "Purchase" Then
95 MsgBox "The order # Entered is a Purchase Order. Please enter the Sales Order
    # again"
96 Exit Sub
97 End If
98 Application.ScreenUpdating = False
99 .Range("O7").Value = Orders.Range("B" & OrderRow).Value 'Date & time
100 End With
101 'Load Items
102 With SalesItems
103 .Range("R3:AA9999").ClearContents 'Clear previous data
104 LastRow = .Range("A99999").End(xlUp).Row
105 If LastRow < 4 Then Exit Sub
106 .Range("A3:L" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
    "P2:P3"), CopyToRange:=.Range("R2:AA2"), Unique:=True
107 LastResultRow = .Range("R99999").End(xlUp).Row
108 If LastResultRow < 3 Then Exit Sub
109 For ResultRow = 3 To LastResultRow

```

1 2 3

```

1 2 3
110 ItemRow = .Range("Y" & ResultRow).Value 'Item Row
111 Sales.Range("J" & ItemRow & ":O" & ItemRow).Value = .Range("R" & ResultRow
    & ":W" & ResultRow).Value 'Item Details
112 Sales.Range("P" & ItemRow).Formula = "=M" & ItemRow & "*O" & ItemRow
    'Total Formula
113 Sales.Range("Q" & ItemRow & ":R" & ItemRow).Value = .Range("Z" & ResultRow
    & ":AA" & ResultRow).Value 'Sales & Purch DB Rows
114 Next ResultRow
115 End With
116 UpdateOrderTotals 'Run Macro To Add/Update Totals
117 Application.ScreenUpdating = True
118 End Sub
119
120 Sub SalesOrder_Print()
121 With Sales
122 LastRow = .Range("K99999" ).End(xlUp).Row + 3
123 .PageSetup.PrintArea = "$K$4:$P$" & LastRow
124 .PrintOut , , , True, , , False
125 End With
126 End Sub
127
128 Sub SalesOrder_SaveUpdate()
129 With Sales
130 LastItemRow = .Range("K99999" ).End(xlUp).Row
131 If LastItemRow < 9 Then
132 MsgBox "Please make sure to add at least one product before saving"
133 Exit Sub
134 End If
135 If .Range("B2" ).Value = Empty Then 'New Order
136 OrderRow = Orders.Range("A99999" ).End(xlUp).Row + 1
137 Orders.Range("A" & OrderRow).Value = .Range("B4" ).Value 'Order ID
138 Else 'Existing Order
139 OrderRow = .Range("B2" ).Value 'Existing Order DB Row
140 End If
141 Orders.Range("B" & OrderRow).Value = .Range("O7" ).Value 'Order Date
142 Orders.Range("C" & OrderRow).Value = "Sales" 'Set Order Type
143 Orders.Range("D" & OrderRow).Value = .Range("O6" ).Value 'Cashier
144 Orders.Range("E" & OrderRow).Value = .Range("P" & LastItemRow + 3).Value
    'Order Total
145 LastPurchItemDBRow = PurchItems.Range("A99999" ).End(xlUp).Row
146 'Save/Udpate Items
147 For ItemRow = 9 To LastItemRow
148 If .Range("R" & ItemRow).Value = Empty Then 'New Purch Item DB
149 'Run Advanced Filter To Find To Locate Oldest Available Inventory Items
150 PurchQty = .Range("M" & ItemRow).Value
151 With PurchItems
152 .Range("T3" ).Value = Sales.Range("J" & ItemRow).Value 'Set Order ID
    Criteria
153 .Range("A3:O" & LastPurchItemDBRow).AdvancedFilter xlFilterCopy,
    CriteriaRange:=.Range("T2:U3" ), CopyToRange:=.Range("Z2:AB2" ), Unique:
    =True
154 LastResultRow = .Range("AB99999" ).End(xlUp).Row
155 If LastResultRow < 3 Then
156 MsgBox "There are no remaining items for " & Sales.Range("K" &
    ItemRow).Value & vbCrLf & "Please make sure to create a purchase
    item for this product before it can be sold"
157 Exit Sub
158 End If
159 TotalRemainQty = .Range("AC2" ).Value 'Total Remaining Qty Available
160 If PurchQty > TotalRemainQty Then

```

```

161   1 2 3 4 5 6   MsgBox "There are only " & TotalRemainQty & " of " & Sales.Range(
      "K" & ItemRow).Value & ", however this Sales Order contains " &
      PurchQty & ", which is more than is available to be sold" & vbCrLf &
      "Please decrease the Qty purchased or create a Purchase Order for
      this product"
162   Exit Sub
163   End If
164
165   If LastResultRow < 4 Then GoTo NoSort 'No sort on Single Row
166   With .Sort 'Sort Based on Sell By Date (Oldest First)
167     .SortFields.Clear
168     .SortFields.Add Key:=PurchItems.Range("AA3" ), SortOn:=xlSortOnValues
      , Order:=xlAscending, DataOption:=xlSortNormal 'Sort Based on Sell
      By Date
169     .SetRange PurchItems.Range("Z3:AB" & LastResultRow) 'Set Range
170     .Apply 'Apply Sort
171   End With
172
173   NoSort:
174   For ResultRow = 3 To LastResultRow
175     RemainQty = .Range("Z" & ResultRow).Value 'Remaining Qty
176     PurchItemDBRow = .Range("AB" & ResultRow).Value 'Item DB Row
177     If RemainQty < PurchQty Then 'Remaining Qty is less than the
      purchase qauntity, spread over multiple purchases
178       .Range("I" & PurchItemDBRow).Value = .Range("I" & PurchItemDBRow
      ).Value + RemainQty 'Update Sold Qty to Only the remaining Qty
179       PurchQty = PurchQty - RemainQty 'Update Purchase Qty
180     Else 'Remaining qty equal or greater than purchase Qty
181       .Range("I" & PurchItemDBRow).Value = .Range("I" & PurchItemDBRow
      ).Value + PurchQty 'Updated Sold Qty
182       PurchQty = 0 'Product has been fully attributed to purchases)
183     End If
184     Sales.Range("R" & ItemRow).Value = PurchItemDBRow 'Purchase Order
      Item DB Row
185     If PurchQty = 0 Then GoTo Attributed 'Check If Product has been
      fully attributed to purchases)
186   Next ResultRow
187
188   End With
189 End If
190 Attributed:
191 'Add/Update Product To Sales Item DB
192 With SalesItems
193   If Sales.Range("Q" & ItemRow).Value = Empty Then 'New Sales Item Row
194     SalesItemDBRow = .Range("A99999" ).End(xlUp).Row + 1 'First Avail Row
195     .Range("A" & SalesItemDBRow).Value = Sales.Range("O5" ).Value 'Order
      ID
196     .Range("J" & SalesItemDBRow).Value = ItemRow 'Sales Order Row
197     .Range("L" & SalesItemDBRow).Value = "=Row()" 'Set DB Row Formula
198     Sales.Range("Q" & ItemRow).Value = SalesItemDBRow 'Set DB Row
199   Else 'Existing sales Item Row
200     SalesItemDBRow = Sales.Range("Q" & ItemRow).Value 'Existing DB Row
201   End If
202   .Range("B" & SalesItemDBRow).Value = Sales.Range("O7" ).Value 'Order
      Date
203   .Range("C" & SalesItemDBRow & ":I" & SalesItemDBRow).Value = Sales.Range(
      "J" & ItemRow & ":P" & ItemRow).Value 'Save Item Details
204   .Range("K" & SalesItemDBRow).Value = Sales.Range("R" & ItemRow).Value
      'Purchase Item DB Row
205 End With

```

```
206     1 2 3 Next ItemRow
207     NoItems:
208     SalesOrder_ListLoad
209     MsgBox OrderType & " Saved"
210     End With
211 End Sub
```

```

1 Option Explicit
2 Dim ItemShp As Shape, IconShp As Shape
3 Dim LastRow As Long, LastResultRow As Long, ResultRow As Long, ProdID As Long
4 Dim ProdDBRow As Long, TypeRow As Long, LastTypeRow As Long
5 Dim ProdName As String, PicName As String, PicPath As String, PicFolder As String, Size
  As String
6 Dim AddinName As String, TypeName As String
7 Dim LeftPos As Double, TopPos As Double, ItemHeight As Double, ItemWidth As Double
8
9 Sub FoodTypes_Refresh()
10     With Sales
11         For Each ItemShp In Sales.Shapes
12             On Error Resume Next
13             If InStr(ItemShp.Name, "FoodType" ) > 0 Then ItemShp.Delete
14             If InStr(ItemShp.Name, "FoodIcon" ) > 0 Then ItemShp.Delete
15             On Error GoTo 0
16         Next ItemShp
17         LastTypeRow = Admin.Range("B99999" ).End(xlUp).Row
18         LeftPos = .Range("E:E" ).Left 'Set Initial left position
19         TopPos = .Range("E3" ).Top 'Set Initial top position
20         If LastTypeRow < 7 Then Exit Sub
21         For TypeRow = 7 To LastTypeRow
22             .Shapes("SampleType" ).Duplicate.Name = "FoodType" & TypeRow 'Create Food
  Type Shape
23             With .Shapes("FoodType" & TypeRow)
24                 .Left = LeftPos
25                 .Top = TopPos
26                 .Width = Sales.Range("E:E" ).Width
27                 .TextFrame2.TextRange.Text = Admin.Range("B" & TypeRow).Value
28                 .OnAction = "TypeSelect"
29             End With
30
31             On Error Resume Next
32             Set IconShp = Admin.Shapes("Icon" & TypeRow)
33             On Error GoTo 0
34             If IconShp Is Nothing Then GoTo NextType
35             IconShp.Copy
36             .Range("A1" ).Select
37             .Paste
38             Selection.Name = "FoodIcon" & TypeRow
39             With .Shapes("FoodIcon" & TypeRow)
40                 .Left = LeftPos + 3
41                 .Top = TopPos + (Sales.Shapes("FoodType" & TypeRow).Height - .Height) / 2
42                 .OnAction = "TypeSelect"
43             End With
44             TopPos = TopPos + .Shapes("FoodType" & TypeRow).Height + 5 'Increase top
  Position
45             NextType:
46
47         Next TypeRow
48     End With
49 End Sub
50 Sub LoadProducts()
51     'Clear all existing Products within the screen
52     For Each ItemShp In Sales.Shapes
53         On Error Resume Next
54         If InStr(ItemShp.Name, "ProdShp" ) > 0 Then ItemShp.Delete
55         If InStr(ItemShp.Name, "ProdPic" ) > 0 Then ItemShp.Delete
56         On Error GoTo 0
57     Next ItemShp

```

1


```

1
58 Sales.Range("B7" ).ClearContents 'Clear out selected Prod ID
59 With Products
60 LastRow = .Range("A99999" ).End(xlUp).Row 'LastRow
61 If LastRow < 4 Then Exit Sub
62 .Range("A3:I" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
'M2:M3" ), CopyToRange:=.Range("Q2:S2" ), Unique:=True
63 LastResultRow = .Range("S99999" ).End(xlUp).Row
64 If LastResultRow < 3 Then Exit Sub
65 Application.ScreenUpdating = False
66 Application.EnableEvents = False
67 PicFolder = [PictureFolder] 'Picture Folder
68 TopPos = Sales.Range("G3" ).Top 'Initial Top Position
69 LeftPos = Sales.Range("G3" ).Left 'Initial Left position
70 ItemWidth = Sales.Range("G:G" ).Width - 5 'Set Shape Width
71 ItemHeight = 50 'Set Item Height
72 For ResultRow = 3 To LastResultRow
73 ProdID = .Range("Q" & ResultRow).Value 'Item ID
74 ProdName = .Range("R" & ResultRow).Value 'Item Name
75 PicName = .Range("S" & ResultRow).Value 'Item Picture
76 Sales.Shapes("SampleProduct" ).Duplicate.Name = "ProdShp" & ProdID 'New
Shape Name
77 With Sales.Shapes("ProdShp" & ProdID)
78 .Top = TopPos
79 .Left = LeftPos 'Set Left Postion
80 .Width = ItemWidth
81 .Height = ItemHeight
82 .TextFrame2.TextRange.Text = ProdName 'Shape Text
83 .OnAction = "Prod_Select" 'Macro to run when selected
84 End With
85 PicPath = PicFolder & "\" & PicName 'Full Picture File Path
86 If Dir(PicPath, vbDirectory) = "" Or PicName = "" Then GoTo NoPic
87 Sales.Pictures.Insert(PicPath).Name = "ProdPic" & ProdID
88 With Sales.Shapes("ProdPic" & ProdID)
89 .LockAspectRatio = msoCTrue
90 If .Width > .Height Then .Width = 40 Else .Height = 40
91 .Top = Sales.Shapes("ProdShp" & ProdID).Top + (Sales.Shapes("ProdShp" &
ProdID).Height - .Height) / 2
92 .Left = Sales.Shapes("ProdShp" & ProdID).Left + Sales.Shapes("ProdShp" &
ProdID).Width - .Width - 3 'Place Picture on right section
93 .OnAction = "Prod_Select" 'Macro to run when selected
94 End With
95 NoPic:
96 TopPos = TopPos + Sales.Shapes("ProdShp" & ProdID).Height + 5
97 Next ResultRow
98 Application.ScreenUpdating = True
99 Application.EnableEvents = True
100 End With
101
102 End Sub
103
104 Sub Prod_Select()
105 Dim OrderRow As Long
106 With Sales
107 Application.ScreenUpdating = False
108 .Range("B10" ).ClearContents 'Clear out selected item row and size
109 PicFolder = [PictureFolder] 'Picture Folder
110 ProdID = Replace(Application.Caller, Left(Application.Caller, 7), "" ) 'Product
ID
111 ProdName = .Shapes("ProdShp" & ProdID).TextFrame2.TextRange.Text 'Item Name
112 .Range("B7" ).Value = ProdID 'Set Item ID

```

```

1 2
113 .Calculate
114 ProdDBRow = .Range("B8" ).Value
115 If .Range("B8" ).Value = "" Then
116     MsgBox "Please make sure product has been added to the Products sheet"
117     Exit Sub
118 End If
119 On Error Resume Next
120 OrderRow = .Range("J9:J999" ).Find(ProdID, , xlFormulas, xlWhole).Row
121 On Error Resume Next
122 If OrderRow <> 0 Then 'Update Qty on Existing Row
123     .Range("M" & OrderRow).Value = .Range("M" & OrderRow).Value + 1
124 Else 'New Row
125     .Range("B1" ).Value = True 'Set Order Load To True
126     OrderRow = .Range("K99999" ).End(xlUp).Row + 1 'Set Order Row
127     .Range("J" & OrderRow).Value = ProdID 'Set Product ID
128     .Range("K" & OrderRow).Value = ProdName 'Product Name
129     .Range("L" & OrderRow).Value = Products.Range("C" & ProdDBRow).Value 'Set
        UPC/SKU
130     .Range("M" & OrderRow).Value = 1 ' Set Qty
131     .Range("N" & OrderRow).Value = Products.Range("E" & ProdDBRow).Value 'Unit
132     .Range("O" & OrderRow).Value = Products.Range("H" & ProdDBRow).Value '
        Sales Price
133     .Range("P" & OrderRow).Value = "=M" & OrderRow & "*O" & OrderRow 'Line
        Total Formula
134     .Range("B1" ).Value = False 'Set Order Load To False
135 End If
136 UpdateOrderTotals
137 Application.ScreenUpdating = True
138 End With
139 End Sub
140 Sub TypeSelect()
141     With Sales
142         TypeRow = Replace(Application.Caller, Left(Application.Caller, 8), "" ) 'Food
            Type Row
143         Products.Range("M3" ).Value = .Shapes(Application.Caller).TextFrame2.TextRange.
            Text 'Food Type Name Criteria
144         For Each ItemShp In Sales.Shapes
145             If InStr(ItemShp.Name, "FoodType" ) > 0 Then ItemShp.Fill.ForeColor.RGB = RGB(
                84, 130, 53) 'Set Light Green Color For ALL Categories
146         Next ItemShp
147         Sales.Shapes(Application.Caller).Fill.ForeColor.RGB = RGB(56, 87, 35) 'Select
            Dark Green Color
148         LoadProducts 'Run Macro to load Products
149     End With
150 End Sub
151
152 Sub UpdateOrderTotals()
153     With Sales
154         LastRow = .Range("K99999" ).End(xlUp).Row
155         .Range("O" & LastRow + 1 & ":P" & LastRow + 999).ClearContents 'Clear and
            Previous totals
156         .Range("FooterRng" ).Copy
157         .Range("O" & LastRow + 1).PasteSpecial xlPasteValues
158         .Range("P" & LastRow + 1).Formula = "=Sum(P9:P" & LastRow & ")" 'Subtotal
159         .Range("P" & LastRow + 2).Formula = "=IF(SalesTaxOpt=" & Chr(34) & "No" & Chr(
            34) & ",0,P" & LastRow + 1 & "*SalesTaxRate)" 'Tax Formula
160         .Range("P" & LastRow + 3).Formula = "=Sum(P" & LastRow + 1 & ":P" & LastRow +
            2 & ")" 'Total Formula
161         .Range("K9" ).Select
162         Application.CutCopyMode = False

```

```
163 | 1 2  
164 | End With  
    | End Sub
```

A

Activate, 15, 17
 ActiveSheet, 15
 Add, 14, 17, 19, 20, 23, 24, 26
 AddinName, 28
 Admin, 14, 28
 Admin_AddIcon, 14
 Admin_ClearIcon, 14
 AdvancedFilter, 17, 19, 20, 24, 25, 29
 AllowMultiSelect, 14
 Application, 14, 15, 20, 21, 24, 25, 29, 30
 Apply, 17, 19, 20, 23, 24, 26
 Attributed, 26

B

BrowseForPicFolder, 14

C

Calculate, 30
 Caller, 15, 29, 30
 Chr, 5, 17, 30
 Clear, 17, 19, 20, 23, 24, 26
 ClearContents, 5, 8, 10, 14, 17, 19, 20, 23, 24, 29, 30
 Copy, 28, 30
 CopyToRange, 17, 19, 20, 24, 25, 29
 CountLarge, 3, 5, 8, 10
 CriteriaRange, 17, 19, 20, 24, 25, 29
 CutCopyMode, 30

D

Dashboard, 16
 Dashboard_Refresh, 16
 DataOption, 17, 19, 20, 23, 24, 26
 Delete, 14, 16, 19, 23, 28
 Dir, 16, 29
 Duplicate, 28, 29

E

Empty, 3, 5, 8, 10, 16-21, 23-26
 EnableEvents, 29
 EntireRow, 19, 23
 Explnv, 16, 17
 Explnv_GoToPO, 17
 Explnv_Refresh, 5, 16-18
 Explnv_RemoveItems, 17
 Explicit, 3-14, 16, 17, 19, 23, 28

F

FileDialog, 14
 Fill, 30
 Filters, 14
 Find, 8, 10, 20, 24, 30
 FoodTypes_Refresh, 28
 ForeColor, 30
 Formula, 25, 30

H

Height, 14, 16, 28, 29

I

IconName, 14
 IconPic, 14
 IconShp, 28
 Insert, 14, 16, 29
 InStr, 16, 28, 30
 Intersect, 3, 5, 8, 10
 InvCount, 17, 18
 IsNumeric, 18

ItemDBRow, 19, 21, 22

ItemHeight, 28, 29

ItemRow, 17-23, 25-27

ItemShp, 28, 30

ItemWidth, 28, 29

K

Key, 17, 19, 20, 23, 24, 26

L

LastItemRow, 19, 21, 23, 25
 LastPurchItemDBRow, 23, 25
 LastResultRow, 17, 19, 20, 23-26, 28, 29
 LastRow, 5, 17-21, 23-25, 28-30
 LastTypeRow, 28
 Left, 3, 14, 16, 28-30
 LeftPos, 16, 28, 29
 LoadProducts, 28, 30
 LockAspectRatio, 14, 16, 29

M

MainMenu, 14
 MsgBox, 8, 10, 17-27, 30
 msoCTrue, 3, 14, 16, 29
 msoFalse, 3
 msoFileDialogFilePicker, 14
 msoFileDialogFolderPicker, 14

N

Name, 14, 16, 28-30
 NameType, 19, 23
 NextProd, 16
 NextType, 28
 NoItems, 22, 23, 27
 NoPic, 29
 NoPicFolder, 16
 NoSelection, 14
 NoSort, 26
 NotSaved, 19, 23
 Now, 19, 23

O

OnAction, 28, 29
 Order, 17, 19, 20, 23, 24, 26
 OrderID, 19, 20, 23, 24
 OrderRow, 19-21, 23-25, 29, 30
 Orders, 19-21, 23-25
 OrderType, 19, 20, 22-24, 27

P

PageSetup, 21, 25
 Paste, 28
 PasteSpecial, 30
 PicFolder, 14, 16, 28, 29
 PicName, 16, 28, 29
 PicPath, 14, 16, 28, 29
 PictureFolder, 16, 29
 Pictures, 14, 16, 29
 POBRow, 17, 18
 PONumb, 17
 PrintArea, 21, 25
 PrintOut, 21, 25
 Prod_Select, 29
 ProdDBRow, 28, 30
 ProdID, 8, 10, 19, 21, 28-30
 ProdName, 28-30
 ProdPic, 16
 ProdRow, 8, 10, 16
 Products, 8, 10, 29, 30
 Purchases, 17, 19-21

PurchItemDBRow, 23, 26

PurchItems, 17-23, 25, 26

PurchOrder_AddNew, 19

PurchOrder_Delete, 19

PurchOrder_ListLoad, 19, 22

PurchOrder_Load, 8, 20

PurchOrder_Print, 21

PurchOrder_SaveUpdate, 21

PurchQty, 23, 25, 26

R

Range, 3, 5, 8, 10, 14, 16-26, 28-30
 RemainQty, 17, 23, 26
 RemoveQty, 17
 Replace, 29, 30
 ResultRow, 19, 20, 23-26, 28, 29
 RGB, 30
 Row, 3, 5, 8, 10, 17, 19-21, 23-26, 28-30

S

Sales, 23-26, 28-30
 SalesItemDBRow, 23, 26
 SalesItems, 23, 24, 26
 SalesOrder_AddNew, 23
 SalesOrder_Delete, 23
 SalesOrder_ListLoad, 23, 27
 SalesOrder_Load, 10, 24
 SalesOrder_Print, 25
 SalesOrder_SaveUpdate, 25
 ScreenUpdating, 20, 21, 24, 25, 29, 30
 SelectedItems, 14
 Selection, 28
 SelRow, 14, 17
 SetRange, 17, 19, 20, 23, 24, 26
 Shape, 16, 28
 Shapes, 3, 14-16, 28-30
 Sheets, 15
 Show, 14
 ShtName, 15
 Size, 28
 SkipSort, 20, 24
 Sort, 17, 19, 20, 23, 24, 26
 SortFields, 17, 19, 20, 23, 24, 26
 SortOn, 17, 19, 20, 23, 24, 26

T

Target, 3, 5, 8, 10
 Text, 15, 28-30
 TextFrame2, 15, 28-30
 TextRange, 15, 28-30
 Title, 14
 Top, 3, 14-16, 28, 29
 TopPos, 16, 28, 29
 TotalRemainQty, 23, 25, 26
 TypeName, 28
 TypeRow, 28, 30
 TypeSelect, 30

U

Unique, 17, 19, 20, 24, 25, 29
 UpdateOrderTotals, 25, 30

V

Value, 3, 5, 8, 10, 14, 16-26, 28-30
 vbCrLf, 25, 26
 vbDirectory, 16, 29
 vbNo, 19, 23
 vbYesNo, 19, 23
 Visible, 3

W

Width, [14](#), [16](#), [28](#), [29](#)

Worksheet_Change, [5](#), [8](#), [10](#)

Worksheet_SelectionChange, [3](#), [5](#), [8](#), [10](#)

X

xlAscending, [17](#), [19](#), [23](#), [26](#)

xlDescending, [20](#), [24](#)

xlFilterCopy, [17](#), [19](#), [20](#), [24](#), [25](#), [29](#)

xlFormulas, [30](#)

xlPasteValues, [30](#)

xlSortNormal, [17](#), [19](#), [20](#), [23](#), [24](#), [26](#)

xlSortOnValues, [17](#), [19](#), [20](#), [23](#), [24](#), [26](#)

xlUp, [5](#), [17](#), [19-21](#), [23-26](#), [28-30](#)

xlValues, [8](#), [10](#), [20](#), [24](#)

xlWhole, [8](#), [10](#), [20](#), [24](#), [30](#)

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