

VBA SOURCE CODE BOOK

INVOICE & PAYMENT SYSTEM - UPDATED

**EXCLUSIVE FOR
SILVER & GOLD
MEMBERS**

INVOICE PAYMENTS MANAGER

Save Payment Add New Delete Payment Prev. Next

Customer Dolores Richman Payment Date 1/5/2023

Payment Type Bank Transfer Pmnt. Amount \$5,066.22

Payment Notes Final Payment Due in 30 Days

Total Applied \$5,066.22

Apply	Inv. Date	Inv. #	Invoice Amount	Prev. Payments	Balance	Amount Applied
✓	5/18/2022	2	\$1,653.73	\$0.00	\$1,653.73	\$1,653.73
✓	12/28/2022	5	\$3,251.61	\$0.00	\$3,251.61	\$3,251.61
✓	1/4/2023	8	\$34.63	\$0.00	\$34.63	\$34.63
	1/19/2023	9	\$26.25	\$0.00	\$26.25	\$26.25
	1/17/2023	10	\$262.50	\$0.00	\$262.50	\$100.00
	1/5/2023	11	\$445.20	\$0.00	\$445.20	

**FEATURE, FIX,
OR FOCUS**



Excel For Freelancers

How To Create Your Own Invoice Payment System In Excel - FROM SCRATCH



[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 400,000 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



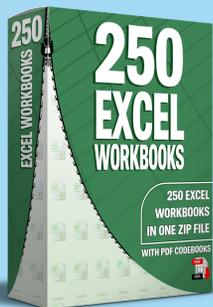
This comprehensive program will take you through a 12-phase process that will turn your enthusiasm for Excel into passive income.

[Click here to learn more](#)



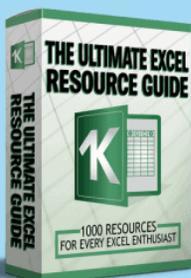
16 hour masterclass that will teach you the tips, tricks and techniques on how to create a dynamic single-click dashboard, and a ton more

[Click here to learn more](#)



Incredible Package of 250 of my BEST Applications now with PDF VBA Codebooks packed into a SINGLE ZIP File which also includes the "250 Workbook Library".

[Click here to learn more](#)



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

[Click here to learn more](#)

Table of Contents

Projects.....	3
VBAProject.....	3
Documents.....	3
Admin.....	3
(Declarations).....	3
Worksheet_SelectionChange [Sub].....	3
Customers.....	4
(Declarations).....	4
Invoice.....	5
(Declarations).....	5
Worksheet_Change [Sub].....	5
Worksheet_SelectionChange [Sub].....	6
InvoiceItems.....	10
(Declarations).....	10
InvoiceList.....	11
(Declarations).....	11
Items.....	12
(Declarations).....	12
PayItems.....	13
(Declarations).....	13
PaymentList.....	14
(Declarations).....	14
Payments.....	15
(Declarations).....	15
Worksheet_Change [Sub].....	15
Worksheet_SelectionChange [Sub].....	15
Sheet1.....	16
(Declarations).....	16
ThisWorkbook.....	17
(Declarations).....	17
Forms.....	18
AddCustForm.....	18
(Declarations).....	18
CancelBtn_Click [Sub].....	18
SaveBtn_Click [Sub].....	18
Modules.....	19
Admin_Menu_Macros.....	19
(Declarations).....	19
AppEvents_Start [Sub].....	19
AppEvents_Stop [Sub].....	19
Company_AddLogo [Sub].....	19
EditMode [Sub].....	19
CustCredit_Macros.....	20
(Declarations).....	20
Customer_CreateCredit [Sub].....	20
Email_Macros.....	21
(Declarations).....	21
Email_Summary [Sub].....	21
Invoice_Macros.....	22
(Declarations).....	22
Customer_AddUpdate [Sub].....	22
Invoice_Delete [Sub].....	22
Invoice_Load [Sub].....	22
Invoice_New [Sub].....	23
Invoice_NewPage [Sub].....	23
Invoice_NextInvoice [Sub].....	23
Invoice_NextPage [Sub].....	24
Invoice_PageLoad [Sub].....	24
Invoice_PrevInvoice [Sub].....	24
Invoice_PrevPage [Sub].....	25
Invoice_SavedMsg [Sub].....	25
Invoice_SaveUpdate [Sub].....	25
Payments_Macros.....	27
(Declarations).....	27
Payments_AddNew [Sub].....	27
Payments_Delete [Sub].....	27

Table of Contents

Payments_Load [Sub].....	27
Payments_LoadOpenInvoices [Sub].....	28
Payments_Next [Sub].....	28
Payments_Previous [Sub].....	29
Payments_SaveUpdate [Sub].....	29
PopUp_Calendar.....	31
(Declarations)	31
CalCol [Sub]	31
CalendarHide [Sub].....	31
CalendarShow [Sub]	31
CalFormulaReplacement [Sub]	32
CheckForSheet [Sub].....	32
CreateCalSht [Sub].....	33
GroupCal [Sub].....	35
MacroLinkRemover [Sub]	35
NextMonth [Sub]	36
NextYear [Sub]	36
PrevMonth [Sub]	36
PrevYear [Sub]	37
ReplaceCalendar [Sub].....	37
SelectDay [Sub]	37
ShowSettings [Sub]	37
UnGroupCal [Sub]	37
Print_Macros.....	39
(Declarations)	39
Invoice_Print [Sub].....	39
Test [Sub]	39

```
1 Option Explicit
2
3 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
4     If Target.CountLarge > 1 Then Exit Sub
5
6     'On Default Invoice Status Selection
7     If Not Intersect(Target, Range("F6:F11" )) Is Nothing And Range("E" & Target.Row).
8         Value <> Empty Then
9         Range("F6:F11" ).ClearContents
10        Target.Value = Chr(252)
11        Range("E6" ).Select
12    End If
13
14    'On Default Payment Terms Selection
15    If Not Intersect(Target, Range("J6:J11" )) Is Nothing And Range("H" & Target.Row).
16        Value <> Empty Then
17        Range("J6:J11" ).ClearContents
18        Target.Value = Chr(252)
19        Range("H6" ).Select
20    End If
21 End Sub
```

1 Option Explicit

2

```

1 Option Explicit
2 Dim ItemTypeRow As Long, ItemCol As Long, ItemRow As Long, TermRow As Long
3 Private Sub Worksheet_Change(ByVal Target As Range)
4     Dim ItemDBRow As Long, InvItemRow As Long, SelRow As Long
5     If Target.CountLarge > 1 Then Exit Sub
6
7     'On Invoice Search
8     If Not Intersect(Target, Range("F2" )) Is Nothing And Range("F2" ).Value <> Empty
9         Then
10            If Range("B2" ).Value = Empty Then 'Incorrect Invoice #
11                MsgBox "Please select a correct Invoice #"
12                Exit Sub
13            End If
14            Range("B4" ).Value = Range("F2" ).Value 'Set Invoice #
15            Range("F2" ).ClearContents
16            Invoice_Load
17        End If
18
19        'On Item Name Change Load Item Details (But Not On Page/Invoice Load)
20        If Not Intersect(Target, Range("K10:K44" )) Is Nothing And Range("B1" ).Value =
21            False Then
22                'Check For Invoice Save
23                If Range("B5" ).Value = Empty Then 'Invoice Not Save
24                    MsgBox "Please make sure to save the Invoice before adding items"
25                    Application.EnableEvents = False
26                    Target.ClearContents
27                    Application.EnableEvents = True
28                    Exit Sub
29                End If
30                If Target.Value <> Empty Then 'Non Empty
31                    On Error Resume Next
32                    ItemDBRow = Items.Range("Item_Name" ).Find(Target.Value, , xlValues, xlWhole).
33                    Row
34                    On Error GoTo 0
35                    If ItemDBRow = 0 Then
36                        MsgBox "Please select a correct item from the drop down list"
37                        Exit Sub
38                    End If
39                    Range("L" & Target.Row).Value = Items.Range("B" & ItemDBRow).Value
40                    'Description
41                    Range("M" & Target.Row).Value = Items.Range("C" & ItemDBRow).Value 'Default
42                    Qty
43                    Range("N" & Target.Row).Value = Items.Range("D" & ItemDBRow).Value
44                    'Amount/Price
45                    Range("P" & Target.Row).Value = Items.Range("E" & ItemDBRow).Value 'Taxable
46                Else 'Empty Clear Row
47                    Range("L" & Target.Row & ":N" & Target.Row).ClearContents 'Clear Item
48                    Details
49                    Range("P" & Target.Row).ClearContents 'Clear Tax
50                End If
51            End If
52        End If
53
54        'On Any Invoice Item Detail Change, Save To Database (not on Invoice Load)
55        If Not Intersect(Target, Range("J10:P44" )) Is Nothing And Range("B1" ).Value =
56            False Then
57            SelRow = Target.Row 'Set Selected Row
58            'Update Invoice Items DB
59            If Range("Q" & SelRow).Value = Empty Then 'New Invoice Item

```

```

1 2 3
52  InvItemRow = InvoiceItems.Range("A9999").End(xlUp).Row + 1 'First Avail.
    Row
53  InvoiceItems.Range("A" & InvItemRow).Value = Range("B4").Value 'Invoice
    Number
54  InvoiceItems.Range("B" & InvItemRow).Value = Range("B11").Value 'Page
    Number
55  InvoiceItems.Range("J" & InvItemRow).Value = SelRow 'Set Invoice Row
56  InvoiceItems.Range("K" & InvItemRow).Value = "=row()" 'Set Row Formula
57  Range("Q" & Target.Row).Value = InvItemRow 'Set Item Row
58  Else 'Existing Invoice Item
59  InvItemRow = Range("Q" & Target.Row).Value 'Existing Item Row
60  End If
61  InvoiceItems.Cells(InvItemRow, Target.Column - 7).Value = Target.Value
62  InvoiceItems.Range("H" & InvItemRow).Value = Range("O" & SelRow).Value 'Update
    Total
63  End If
64
65  'On Any Terms Or Inv. Date Change, Update Due Date (not on Invoice Load) (if
    applicable)
66  If Not Intersect(Target, Range("E4,E6")) Is Nothing And Range("E4").Value <> Empty
    And Range("E6").Value <> Empty And Range("B1").Value = False Then
67  On Error Resume Next
68  TermRow = Admin.Range("Terms").Find(Range("E6").Value, , xlValues, xlWhole).Row
69
    On Error GoTo 0
70  If TermRow = 0 Then Exit Sub 'Exit if not found
71  Range("E8").Value = Range("E4").Value + Admin.Range("I" & TermRow).Value
    'Update Due Date
72  End If
73
74
75  '''Customize Invoice Changes
76  If Range("B10").Value = False Then Exit Sub 'Exit unless in Edit mode
77  Dim ItemLblShp As Shape
78  'On Change of Show or Label
79  If Not Intersect(Target, Range("F13:G22")) Is Nothing And Range("B8").Value =
    False And Range("E" & Target.Row).Value <> Empty Then
80  ItemTypeRow = Range("B9").Value 'Selected Item Type Row
81  ItemCol = 22 + ((ItemTypeRow - 13) * 4) + Target.Column 'Column to Save To
82  Cells(Target.Row, ItemCol).Value = Target.Value 'Update Data table
83  'On Change of Label, Show/Hide Label (but not on Item Detail Columns)
84  If Not Intersect(Target, Range("G13:G22")) Is Nothing And Range("B9").Value <>
    16 Then
85  On Error Resume Next
86  Set ItemLblShp = Shapes(Range("E" & Target.Row).Value & "_Lb1")
87  If Target.Value = Empty Then ItemLblShp.Visible = msoFalse Else ItemLblShp.
    Visible = msoCTrue 'Hide/Show Label based on Label Value
88  End If
89  End If
90
91  End Sub
92
93  Private Sub Worksheet_SelectionChange(ByVal Target As Range)
94  Dim ItemNm As String
95  Dim ItemShp As Shape, ItemLblShp As Shape, GrpShp As Shape, Cal As Shape
96  If Target.CountLarge > 2 Then Exit Sub
97  On Error Resume Next
98  Set Cal = Shapes("Calendar")
99  On Error GoTo 0
100 If Cal Is Nothing Then ReplaceCalendar 'Shape Deleted

```

```

1
101 If Shapes("Calendar").Visible = True Then CalendarHide 'Run macro to hide calendar
102
103 If Not Intersect(Target, Range("E4,E8,J10:J44")) Is Nothing Then 'Change this to
    any cell (or cells) you would like to have the Pop-Up Calendar Appear
104     CheckForSheet
105     CalendarShow
106 Else:
107     CheckForSheet
108     CalendarHide
109 End If
110
111 'On Tax Selection Change but not on Invoice Load
112 On Error Resume Next
113 If Not Intersect(Target, Range("P10:P44")) Is Nothing And Range("B1").Value =
    False Then
114     If Target.Value = Chr(252) Then Target.ClearContents Else Target.Value = Chr(252)
115     Range("P8").Select 'Select another cell
116 End If
117
118
119 '''Customize Invoice Changes
120
121 If Range("B10").Value = False Then Exit Sub 'Exit unless in Edit mode
122 'On Edit Invoice Item Menu Select
123 If Not Intersect(Target, Range("D13:D16")) Is Nothing Then
124     Range("B8").Value = True 'Set Menu Change to True
125     Range("B9").Value = Target.Row 'Set Type /Trigger CF
126     Range("E13:G22").ClearContents 'Clear Fields
127     ItemCol = 27 + ((Target.Row - 13) * 4)
128     Range("E13:G22").Value = Range(Cells(13, ItemCol), Cells(22, ItemCol + 2)).Value
    'Load Item Details
129     Range("B8").Value = False 'Set Menu Change To False
130 End If
131
132 'On Show/Hide Selection
133 If Not Intersect(Target, Range("F13:F22")) Is Nothing And Range("E" & Target.Row).
    Value <> Empty Then
134     Dim InvWidth As Double, DescWidth As Double
135     Set ItemShp = Nothing
136     Set ItemLblShp = Nothing
137     If Target.Value = Chr(252) Then Target.ClearContents Else Target.Value = Chr(252)
138
139     ItemTypeRow = Range("B9").Value 'Set Item Type Row
140     ItemRow = Target.Row
141     'If Item Details are selected, then Hide/Show Columns
142     If ItemTypeRow = 16 Then 'Item Details (Columns)
143         ItemCol = Target.Row - 3 'Set Item Column
144         If Target.Value = "" Then 'Hide Column
145             Cells(1, ItemCol).EntireColumn.Hidden = True
146         Else 'Show Column
147             Cells(1, ItemCol).EntireColumn.Hidden = False
148         End If
149         'Adjust Item Description column if its not hidden
150         If ItemCol <> 12 Then
151             InvWidth = Columns("J").ColumnWidth + Columns("K").ColumnWidth + Columns(
                "L").ColumnWidth + Columns("M").ColumnWidth +
                Columns("N").ColumnWidth + Columns("O").ColumnWidth + Columns("P").
                ColumnWidth

```

1 2 3 4

```

1 2 3 4
152 DescWidth = Columns("L").ColumnWidth
153 If InvWidth > 546.5 Then Columns("L").ColumnWidth = (DescWidth - (InvWidth
- 93.6))
154 If InvWidth < 546.5 Then Columns("L").ColumnWidth = (DescWidth + (93.6 -
InvWidth))
155 End If
156
157 Else 'Invoice Fields (Shapes)
158 ItemCol = 29 + ((ItemTypeRow - 13) * 4)
159 ItemNm = Range("E" & Target.Row).Value 'Set Item Name
160
161 'Check For Item Value Shape
162 On Error Resume Next
163 Set ItemShp = Shapes(ItemNm)
164 On Error GoTo 0
165 If ItemShp Is Nothing Then 'Create & Position Value Shape if it does not
exist
166 If ItemNm <> "Logo" Then
167 Shapes("SampleInvShp").Duplicate.Name = ItemNm 'Create Shape If
missing
168 Else 'Logo
169 If Admin.Range("C10").Value = Empty Or Dir(Admin.Range("C10").Value,
vbDirectory) = "" Then GoTo CheckForLbl
170 On Error Resume Next
171 Pictures.Insert(Admin.Range("C10").Value).Name = ItemNm 'Set Logo
Picture
172 On Error GoTo 0
173 End If
174 Set ItemShp = Shapes(ItemNm)
175 With ItemShp
176 If ItemNm <> "Logo" Then .DrawingObject.Formula = "=" & Cells(ItemRow,
ItemCol + 1).Address
177 .Left = Range("L1").Left
178 .Top = Range("L5").Top
179 .Placement = xlMove 'Move but not size with cells (columns)
180 End With
181 End If
182 CheckForLbl:
183 'Check For Item Label Shape
184 On Error Resume Next
185 Set ItemLblShp = Shapes(ItemNm & "_Lbl")
186 On Error GoTo 0
187 If ItemLblShp Is Nothing Then 'Create & Position Value Shape if it does not
exist
188 Shapes("SampleInvShp").Duplicate.Name = ItemNm & "_Lbl" 'Create Shape If
missing
189 Set ItemLblShp = Shapes(ItemNm & "_Lbl")
190 With ItemLblShp
191 .DrawingObject.Formula = "=" & Cells(ItemRow, ItemCol).Address
192 .Left = Range("L1").Left - 50
193 .Top = Range("L5").Top
194 .Placement = xlMove
195 End With
196 End If
197
198 'Group Shapes if group does not exist
199 On Error Resume Next
200 Set GrpShp = Shapes(ItemNm & "_Grp")
201 On Error GoTo 0
202 If GrpShp Is Nothing Then 'Group Shapes

```

```
1 2 3 4
203 Shapes.Range(Array(ItemShp.Name, ItemLblShp.Name)).Group.Name = ItemNm &
    "_Grp"
204 Shapes(ItemNm & "_Grp" ).Placement = xlMove 'Set to Move but not size with
    cells
205 End If
206
207 If Target.Value = Empty Or Target.Offset(0, 1).Value = Empty Then 'Ungroup
    and Hide on Empty Label
208     On Error Resume Next
209     Shapes(ItemNm & "_Grp" ).Ungroup
210     On Error Resume Next
211     ItemLblShp.Visible = msoFalse
212 Else
213     ItemLblShp.Visible = msoCTrue 'Show/Hide Invoice Label
214 End If
215 If Target.Value = Empty Then 'Hidden
216     ItemShp.Visible = msoFalse
217 Else 'Visible
218     ItemShp.Visible = msoCTrue 'Show/Hide Invoice Field Value
219 End If
220 End If
221 Range("F2" ).Select
222 End If
223 End Sub
```

1	Option Explicit
2	

1	Option Explicit
2	

1

Option Explicit

1	Option Explicit
2	

1 Option Explicit

2

```
1 Option Explicit
2
3 Private Sub Worksheet_Change(ByVal Target As Range)
4     'On Customer Change for new payments, not on customer load,
5     If Not Intersect(Target, Range("F3" )) Is Nothing And Range("B2" ).Value = False _
6     And Range("B4" ).Value = Empty And Range("F3" ).Value <> "" Then
7         Payments_LoadOpenInvoices
8
9     'On Applied Payment Amount change but not on payment load then select/Unselect
10    apply
11    If Not Intersect(Target, Range("J11:J99" )) Is Nothing And Range("E" & Target.Row).
12    Value <> Empty And Range("B2" ).Value = False Then
13        If Target.Value = Empty Then Range("D" & Target.Row).ClearContents Else Range(
14        "D" & Target.Row).Value = Chr(252)
15    End If
16 End Sub
17
18 Private Sub Worksheet_SelectionChange(ByVal Target As Range)
19     If Target.CountLarge > 1 Then Exit Sub
20     If Not Intersect(Target, Range("D11:D999" )) Is Nothing And Range("E" & Target.Row)
21     .Value <> Empty Then
22         If Target.Value = "" Then Target.Value = Chr(252) Else Target.ClearContents
23         'Check For Applied Above Payment Amount
24         If Target.Value = Chr(252) Then 'Payment applied
25             Range("J" & Target.Row).Value = Range("I" & Target.Row).Value 'Set Amt.
26             Applied = Balance
27         Else 'Clear Applied Payment
28             Range("J" & Target.Row).ClearContents 'Clear Applied Amount
29         End If
30     End If
31     Range("F3" ).Select
32 End Sub
33 End Sub
```

1 Option Explicit

2

1 Option Explicit

2

```
1 Option Explicit
2
3 Private Sub CancelBtn_Click()
4     AddCustForm.Hide
5 End Sub
6
7 Private Sub SaveBtn_Click()
8     Dim CustomerFld As Control
9     Dim CustRow As Long, CustCol As Long
10    If Me.Field1.Value = Empty Then
11        MsgBox "Please make sure to save a Customer Name before saving"
12        Exit Sub
13    End If
14    With Customers
15        If Invoice.Range("B3").Value = Empty Then 'New Customer
16            CustRow = .Range("A99999").End(xlUp).Row + 1 'First avail Row
17            On Error Resume Next
18            .Range("A" & CustRow).Value = Application.WorksheetFunction.Max(.Range(
19                "Cust_ID" )) + 1
20            On Error GoTo 0
21            If .Range("A" & CustRow).Value = Empty Then .Range("A" & CustRow).Value = 1
22            'Set First Cust ID to 1
23        Else
24            CustRow = Invoice.Range("B3").Value
25        End If
26        For CustCol = 2 To 8
27            Set CustomerFld = Me.Controls("Field" & CustCol - 1)
28            .Cells(CustRow, CustCol).Value = CustomerFld.Value
29        Next CustCol
30        AddCustForm.Hide
31        Invoice.Range("E5").Value = Field1.Value 'Set Customer Name
32    End With
33 End Sub
```

```
1 Option Explicit
2
3 Sub AppEvents_Start()
4     Application.EnableEvents = True
5     Application.Calculation = xlCalculationAutomatic
6     Application.ScreenUpdating = True
7 End Sub
8
9
10 Sub AppEvents_Stop()
11     Application.EnableEvents = False
12     Application.Calculation = xlCalculationManual
13     Application.ScreenUpdating = False
14 End Sub
15
16 Sub Company_AddLogo()
17     Dim LogoPic As FileDialog, PicPath As String
18     Set LogoPic = Application.FileDialog(msoFileDialogFilePicker)
19     With LogoPic
20         .Title = "Please select a logo picture"
21         .Filters.Add "Picture Files" , "*.jpg;*.png;*.gif" , 1
22         .AllowMultiSelect = False
23         If .Show <> -1 Then GoTo NoSelection
24         PicPath = .SelectedItems(1)
25     End With
26     Admin.Range("C10" ).Value = PicPath
27     NoSelection:
28 End Sub
29
30
31 Sub EditMode()
32     Dim MovInc As Long, BtnMov As Long
33     With Invoice
34         If .Shapes("EditModeBack" ).TextFrame2.TextRange.Text = "Off" Then 'Turn On
35             .Shapes("EditModeSwitch" ).Fill.ForeColor.RGB = RGB(0, 176, 80) ' Turn To
36                 Green Color
37             .Shapes("EditModeBack" ).TextFrame2.TextRange.Text = "On"
38             MovInc = 1
39             .Range("B10" ).Value = True 'Set Edit Mode To True
40         Else 'Turn Off
41             MovInc = -1
42             .Shapes("EditModeSwitch" ).Fill.ForeColor.RGB = RGB(127, 125, 127) 'Turn to
43                 gray color
44             .Shapes("EditModeBack" ).TextFrame2.TextRange.Text = "Off"
45             .Range("B10" ).Value = False 'Set Edit Mode To False
46         End If
47         Application.ScreenUpdating = False
48         For BtnMov = 1 To 29
49             .Shapes("EditModeSwitch" ).Left = .Shapes("EditModeSwitch" ).Left + MovInc
50         Next BtnMov
51         Application.ScreenUpdating = True
52     End With
53 End Sub
```

```
1 Option Explicit
2
3 Sub Customer_CreateCredit()
4     Dim CrAmnt As Double
5     Dim CustNm As String
6     With Payments
7         CrAmnt = .Range("J5").Value - .Range("J9").Value 'Credit Amount
8         CustNm = .Range("F3").Value 'Customer Name
9         Invoice_New
10        Invoice.Range("E5").Value = CustNm
11        Invoice.Range("B6").Value = "Credit" 'Set Terms as Credit
12        Invoice.Shapes("OrderLabel").TextFrame2.TextRange.Text = "CREDIT" 'Set Order
13        Text to Credit
14        Invoice_SaveUpdate 'Save Invoice
15        Invoice.Range("K10").Value = "Credit" 'Set Item To Credit
16        Invoice.Range("N10").Value = -CrAmnt 'Set Credit Amount
17        Invoice_SaveUpdate 'Save Credit
18        Invoice_Print
19        Invoice.Shapes("OrderLabel").TextFrame2.TextRange.Text = "INVOICE" 'Return
20        Order Text to Invoice
21    End With
22 End Sub
```

```

1 Option Explicit
2
3 Sub Email_Summary()
4     Dim OutMail As Object, OutApp As Object
5     Dim PayAmt As Double
6     Dim CustRow As Long, LastRow As Long, InvRow As Long
7     Dim CustNm As String, Subj As String, Mesg As String, AppSumm As String, CustEmail
8     Dim PayDate As Date
9     With Payments
10        If .Range("F3").Value = Empty Or .Range("J5").Value = Empty Then
11            MsgBox "Please make sure to add a Customer Name and Payment Amount"
12            Exit Sub
13        End If
14        LastRow = .Range("E999").End(xlUp).Row
15        If LastRow < 11 Then
16            MsgBox "Please make sure to add at least one invoice for payment"
17            Exit Sub
18        End If
19        CustNm = .Range("F3").Value 'Customer Name
20        On Error Resume Next
21        CustRow = Customers.Range("Cust_Name").Find(CustNm, , xlValues, xlWhole).Row
22        On Error GoTo 0
23        If CustRow <> 0 Then CustEmail = Customers.Range("H" & CustRow).Value 'Customer
24        PayAmt = .Range("J5").Value 'Payment Amount
25        PayDate = .Range("J3").Value 'Payment Date
26        Subj = .Range("M4").Value 'Subject
27        Mesg = .Range("M5").Value 'Message
28        If InStr(Mesg, "[AppliedSummary]") > 0 Then 'Add in Applied Summary
29            For InvRow = 11 To LastRow
30                If .Range("D" & InvRow).Value = Chr(252) Then
31                    AppSumm = AppSumm & "A payment of " & Format(.Range("J" & InvRow).
32                    Value, "$#,##0.00") & _
33                    " has been applied to Invoice #: " & .Range("F" & InvRow).Value & _
34                    ", bringing the Invoice balance to " & Format(.Range("I" & InvRow).
35                    Value, "$#,##0.00") & vbCrLf
36                End If
37            Next InvRow
38        End If
39        Subj = Replace(Replace(Replace(Subj, "[CustName]", CustNm), "[PayDate]",
40        PayDate), "[PayAmt]", Format(PayAmt, "$#,##0.00"))
41        Mesg = Replace(Replace(Replace(Replace(Mesg, "[CustName]", CustNm), "[PayDate]",
42        PayDate), "[PayAmt]", Format(PayAmt, "$#,##0.00")), "[AppliedSummary]",
43        AppSumm)
44        Set OutApp = CreateObject("Outlook.Application")
45        Set OutMail = OutApp.CreateItem(0)
46        With OutMail
47            .to = CustEmail
48            .Subject = Subj
49            .body = Mesg
50            .display 'Use .Send to send without displaying first
51        End With
52    End With
53 End Sub

```

```

1 Option Explicit
2 Dim InvRow As Long, InvItemRow As Long, LastRow As Long, LastItemRow As Long, FieldRow
  As Long, ItemDBRow As Long
3 Dim ResultRow As Long, LastResultRow As Long, ItemRow As Long, TermRow As Long,
  StatusRow As Long, InvNumb As Long
4 Dim HeadShp As Shape
5
6 Sub Customer_AddUpdate()
7     Dim CustRow As Long, CustCol As Long
8     Dim CustomerFld As Control
9     Unload AddCustForm
10    'If Existing Customer is selected, Load Customer Details
11    If Invoice.Range("B3").Value <> Empty Then 'Existing Customer
12        CustRow = Invoice.Range("B3").Value
13        For CustCol = 2 To 8 'Loop Through Customer Columns
14            Set CustomerFld = AddCustForm.Controls("Field" & CustCol - 1)
15            CustomerFld.Value = Customers.Cells(CustRow, CustCol).Value 'Map Data from
              customers into userform
16        Next CustCol
17    End If
18    AddCustForm.Show
19 End Sub
20 Sub Invoice_Delete()
21     If MsgBox("Are you sure you want to delete this Invoice?" , vbYesNo, "Delete
      Invoice" ) = vbNo Then Exit Sub
22     With Invoice
23         AppEvents_Stop
24         If .Range("B5").Value = Empty Then GoTo NotSaved
25         InvRow = .Range("B5").Value 'Order Row
26         InvoiceList.Range(InvRow & ":" & InvRow).EntireRow.Delete 'Delete Invoice row
27         'Remove Invoice Items
28         With InvoiceItems
29             LastRow = .Range("A99999").End(xlUp).Row
30             If LastRow < 3 Then GoTo NotSaved
31             .Range("A3:K" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
              "M2:M3" ), CopyToRange:=.Range("P2:W2" ), Unique:=True
32             LastResultRow = .Range("P99999").End(xlUp).Row
33             If LastResultRow < 3 Then GoTo NotSaved
34             If LastResultRow < 4 Then GoTo SingleRow
35             'Sort based on descending rows
36             With .Sort
37                 .SortFields.Clear
38                 .SortFields.Add Key:=InvoiceItems.Range("W3" ), SortOn:=xlSortOnValues,
              Order:=xlDescending, DataOption:=xlSortNormal
39                 .SetRange InvoiceItems.Range("P3:W" & LastResultRow)
40             .Apply
41             End With
42             SingleRow:
43             For ResultRow = 3 To LastResultRow
44                 ItemDBRow = .Range("W" & ResultRow).Value 'Invoice Item DB Row
45                 .Range(ItemDBRow & ":" & ItemDBRow).EntireRow.Delete 'Delete Invoice Item
              Row
46             Next ResultRow
47             End With
48             NotSaved:
49             Invoice_New 'Clear Out All Invoice Fields
50             AppEvents_Start
51         End With
52     End Sub
53

```

```

54 Sub Invoice_Load()
55     With Invoice
56         If .Range("B5" ).Value = Empty Then
57             MsgBox "Please enter a correct invoice #"
58             Exit Sub
59         End If
60         AppEvents_Stop
61         InvRow = .Range("B5" ).Value 'Invoice Row
62         .Range("B1" ).Value = True 'Set Inv. Load to true
63         .Range("F2,E4:F9,J10:N44,P10:Q44" ).ClearContents
64         .Range("B11" ).Value = 1 'Set Default To Load First Page
65         For FieldRow = 4 To 9
66             .Range("E" & FieldRow).Value = InvoiceList.Cells(InvRow, FieldRow - 2).Value
67             'Fill Invoice Details
68         Next FieldRow
69         .Range("B12" ).Value = InvoiceList.Range("I" & InvRow).Value 'Set Total Pages
70         Invoice_PageLoad 'Load Page Items
71         .Range("B1" ).Value = False 'Set inv. Load to false
72         AppEvents_Start
73     End With
74 End Sub

75 Sub Invoice_New()
76     With Invoice
77         .Range("B1" ).Value = True 'Set Inv. Load to true
78         .Range("B4,F2,E4:F9,J10:N44,P10:Q44" ).ClearContents
79         .Range("B4" ).Value = .Range("B6" ).Value 'Set Next invoice #
80         .Range("E4" ).Value = Date 'Set current Date
81         .Range("B11,B12" ).Value = 1 'Set Selected Page and Total Pages to 1
82         .Range("B1" ).Value = False 'Set inv. Load to false
83         On Error Resume Next
84         TermRow = Admin.Range("J6:J11" ).Find(Chr(252), , xlValues, xlWhole).Row
85         'Default Payment Term Row
86         On Error GoTo 0
87         If TermRow <> 0 Then .Range("E6" ).Value = Admin.Range("H" & TermRow).Value
88         'Set Default Term
89         On Error Resume Next
90         StatusRow = Admin.Range("F6:F12" ).Find(Chr(252), , xlValues, xlWhole).Row
91         'Default Status Row
92         On Error GoTo 0
93         If StatusRow <> 0 Then .Range("E7" ).Value = Admin.Range("E" & StatusRow).Value
94         'Set Default Status
95         If ActiveSheet.Name = "Invoice" Then .Range("E5" ).Select
96     End With
97 End Sub

98 Sub Invoice_NewPage()
99     Invoice_SaveUpdate 'Save New Last page limit
100    Invoice.Range("B11" ).Value = Invoice.Range("B12" ).Value + 1 'Set Selected page to
101    new page
102    Invoice.Range("B12" ).Value = Invoice.Range("B12" ).Value + 1 ' Increment page to +
103    1
104    Invoice_PageLoad
105    Invoice_SaveUpdate 'Save New Last page limit
106 End Sub

107 Sub Invoice_NextInvoice()
108     With Invoice
109         Dim MaxInvNumb As Long
110         On Error Resume Next

```

```

1 2
107 MaxInvNumb = Application.WorksheetFunction.Max(InvoiceList.Range("Invoice_ID" ))
108 On Error GoTo 0
109 InvNumb = .Range("B4" ).Value
110 If MaxInvNumb = 0 Then
111     MsgBox "Please create and save an Invoice first"
112     Exit Sub
113 End If
114 If InvNumb = MaxInvNumb Then
115     MsgBox "You are at the last Invoice"
116     Exit Sub
117 End If
118
119
120 If InvNumb = 0 Or .Range("B5" ).Value = Empty Then 'On New Invoice
121     InvRow = 3 'On Empty Invoice Go to First one created
122 Else 'On Existing Inv. find Next one
123     InvRow = InvoiceList.Range("Invoice_ID" ).Find(InvNumb, , xlValues, xlWhole).
124     Row + 1
125 End If
126 .Range("B4" ).Value = InvoiceList.Range("A" & InvRow).Value
127 Invoice_Load
128 End With
129 End Sub
130
131 Sub Invoice_NextPage()
132 If Invoice.Range("B11" ).Value = Invoice.Range("B12" ).Value Then
133     MsgBox "You are on the last page"
134     Exit Sub
135 End If
136 Invoice.Range("B11" ).Value = Invoice.Range("B11" ).Value + 1 'Increment Page # Up
137 Invoice_PageLoad
138 End Sub
139
140 Sub Invoice_PageLoad()
141 If Invoice.Range("B5" ).Value = Empty Then Exit Sub 'Exit on no invoice
142 Invoice.Range("B1" ).Value = True 'Set Inv. Load to true
143 Invoice.Range("J10:N44,P10:Q44" ).ClearContents 'Clear Existing Page Items
144 With InvoiceItems
145     LastRow = .Range("A9999" ).End(xlUp).Row
146     If LastRow < 4 Then GoTo NoItems
147     .Range("A3:K" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
148     "M2:N3" ), CopyToRange:=.Range("P2:W2" ), Unique:=True
149     LastResultRow = .Range("P9999" ).End(xlUp).Row
150     If LastResultRow < 3 Then GoTo NoItems
151     For ResultRow = 3 To LastResultRow
152         InvItemRow = .Range("P" & ResultRow).Value 'Get Invoice Row
153         Invoice.Range("J" & InvItemRow & ":N" & InvItemRow).Value = .Range("Q" &
154         ResultRow & ":U" & ResultRow).Value 'Item Details
155         Invoice.Range("P" & InvItemRow & ":Q" & InvItemRow).Value = .Range("V" &
156         ResultRow & ":W" & ResultRow).Value 'Tax & DB Row
157     Next ResultRow
158     NoItems:
159 End With
160 Invoice.Range("B1" ).Value = False 'Set inv. Load to false
161 End Sub
162
163 Sub Invoice_PrevInvoice()
164 With Invoice
165     Dim MinInvNumb As Long

```

```

1 2
163 On Error Resume Next
164 MinInvNumb = Application.WorksheetFunction.Min(InvoiceList.Range("Invoice_ID" ))
165 On Error GoTo 0
166 If MinInvNumb = 0 Then
167     MsgBox "Please create and save an Invoice first"
168     Exit Sub
169 End If
170 InvNumb = .Range("B4" ).Value
171 If InvNumb = 0 Or .Range("B5" ).Value = Empty Then 'On New Invoice
172     InvRow = InvoiceList.Range("A9999" ).End(xlUp).Row 'On Empty Invoice Go to
    last one created
173 Else 'On Existing Inv. find Previous one
174     InvRow = InvoiceList.Range("Invoice_ID" ).Find(InvNumb, , xlValues, xlWhole).
    Row - 1
175 End If
176 If .Range("B6" ).Value = 1 Or MinInvNumb = 0 Or MinInvNumb = .Range("B4" ).Value
    Then
177     MsgBox "You are at the first invoice"
178     Exit Sub
179 End If
180 .Range("B4" ).Value = InvoiceList.Range("A" & InvRow).Value 'Place Inv. ID
    inside cell
181 Invoice_Load
182 End With
183 End Sub

184
185 Sub Invoice_PrevPage()
186 If Invoice.Range("B11" ).Value = 1 Then
187     MsgBox "You are on the first page"
188     Exit Sub
189 End If
190 Invoice.Range("B11" ).Value = Invoice.Range("B11" ).Value - 1 'Increment Page #
    Down
191 Invoice_PageLoad
192 End Sub

193 Sub Invoice_SavedMsg()
194 With Invoice.Shapes("InvSavedMsg" )
195     Dim i As Long, Delay As Double, StartTime As Double
196     .Visible = msoCTrue
197     For i = 1 To 100
198         .Fill.Transparency = i / 100
199         Delay = 0.009
200         StartTime = Timer
201         Do
202             DoEvents
203             Loop While Timer - StartTime < Delay
204         Next i
205     .Visible = msoFalse
206 End With
207 End Sub

208 Sub Invoice_SaveUpdate()
209 With Invoice
210     If .Range("B3" ).Value = Empty Then
211         MsgBox "Please make sure to add a customer before saving invoice"
212         Exit Sub
213     End If
214     AppEvents_Stop
215     'Deterermine New Invoice/Existing Invoice
216     If .Range("B5" ).Value = Empty Then 'new Invoice
217         InvRow = InvoiceList.Range("A9999" ).End(xlUp).Row + 1 'First Avail Row

```

1 2 3

```
1 2 3
218 .Range("B4" ).Value = .Range("B6" ).Value 'Next Inv. #
219 InvoiceList.Range("A" & InvRow).Value = .Range("B4" ).Value 'Next Inv. #
220 InvoiceList.Range("J" & InvRow).Value = "=H" & InvRow &
    "-IFERROR(SUMIF(PayItem_InvID,A" & InvRow & ",PayItem_Amount),0)"
221 Else 'Existing Invoice
222     InvRow = .Range("B5" ).Value 'Invoice Row
223 End If
224 For FieldRow = 4 To 9
225     InvoiceList.Cells(InvRow, FieldRow - 2).Value = .Range("E" & FieldRow).Value
    'Add Invoice Details
226 Next FieldRow
227 InvoiceList.Range("H" & InvRow).Value = .Range("AD21" ).Value 'Invoice Total
228 InvoiceList.Range("I" & InvRow).Value = .Range("B12" ).Value 'Total Pages
229 End With
230 AppEvents_Start
231 On Error Resume Next
232 If Application.Caller = "SaveBtn" Then Invoice_SavedMsg 'Run Invoice Fade Out
    Message
233 On Error GoTo 0
234 End Sub
```

```

1 Option Explicit
2 Dim LastRow As Long, LastResultRow As Long, PayRow As Long, PayCol As Long
3 Dim ResultRow As Long, LastPayItemRow As Long, PayItemRow As Long, PayItemDBRow As Long
4
5 Sub Payments_AddNew()
6     Payments.Range("B3,F3:G3,J3,F5:G5,J5,F7:J8,D11:K9999").ClearContents 'Clear Fields
7     Payments.Range("J3").Value = Date 'Set Default Date
8 End Sub
9
10 Sub Payments_Delete()
11     If MsgBox("Are you sure you want to delete this payment?", vbYesNo, "Delete
12         Payment") = vbNo Then Exit Sub
13     With Payments
14         If .Range("B4").Value = Empty Then GoTo NotSaved
15         PayRow = .Range("B4").Value 'Pay Row
16         PaymentList.Range(PayRow & ":" & PayRow).EntireRow.Delete 'Delete Payment Row
17     With PayItems
18         LastRow = .Range("A9999").End(xlUp).Row
19         If LastRow < 4 Then GoTo NotSaved
20         .Range("A3:G" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
21             "J2:J3"), CopyToRange:=.Range("O3:T3"), Unique:=True
22         LastResultRow = .Range("O9999").End(xlUp).Row
23         If LastResultRow < 4 Then GoTo NotSaved
24         If LastResultRow < 5 Then GoTo SkipSort
25     With .Sort
26         .SortFields.Clear
27         .SortFields.Add Key:=PayItems.Range("T4"), SortOn:=xlSortOnValues, Order:=
28             xlDescending, DataOption:=xlSortNormal 'Sort
29         .SetRange PayItems.Range("O4:T" & LastResultRow) 'Set Range
30         .Apply 'Apply Sort
31     End With
32     SkipSort:
33     On Error Resume Next
34     For ResultRow = 4 To LastResultRow
35         PayItemDBRow = .Range("T" & ResultRow).Value 'Pay Item DB Row
36         .Range(PayItemDBRow & ":" & PayItemDBRow).EntireRow.Delete 'Delete Pay
37         Item DB Row
38     Next ResultRow
39 End With
40 NotSaved:
41 Payments_AddNew
42 End With
43 End Sub
44
45 Sub Payments_Load()
46     With Payments
47         If .Range("B4").Value = Empty Then
48             MsgBox "Please make sure to select a correct payment"
49             Exit Sub
50         End If
51         PayRow = .Range("B4").Value 'Payment Row
52         .Range("B2").Value = True 'Payment Load To True
53         .Range("F3:G3,J3,F5:G5,J5,F7:J8,D11:K9999").ClearContents
54         For PayCol = 2 To 6
55             .Range(PaymentList.Cells(1, PayCol).Value).Value = PaymentList.Cells(PayRow,
56                 PayCol).Value 'Load From DB

```

1 2 3

```

1 2 3
54     Next PayCol
55
56     'Load Pay Items
57     With PayItems
58         .Range("M4:T999" ).ClearContents 'Clear Any Previous Result
59         LastRow = .Range("A99999" ).End(xlUp).Row
60         If LastRow < 4 Then GoTo NoData
61         .Range("A3:G" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
62             "J2:J3" ), CopyToRange:=.Range("O3:T3" ), Unique:=True
63         LastResultRow = .Range("O99999" ).End(xlUp).Row
64         If LastResultRow < 4 Then GoTo NoData
65         .Range("M4:N" & LastResultRow).Formula = .Range("M1:N1" ).Formula 'Bring
66             Apply and Inv. Date Formulas
67             .Range("P4:R" & LastResultRow).Formula = .Range("P1:R1" ).Formula ' Inv.
68             Amount, Prev. Payments, Balance
69             Payments.Range("D11:K" & LastResultRow + 7).Value = .Range("M4:T" &
70             LastResultRow).Value 'Bring over pay items
71             NoData:
72             Payments.Range("B2" ).Value = False 'Payment Load To False
73         End With
74     End With
75 End Sub
76
77 Sub Payments_LoadOpenInvoices()
78     Payments.Range("D11:K9999" ).ClearContents 'Clear Any Previous Data
79
80     With Invoicelist
81         LastRow = .Range("A99999" ).End(xlUp).Row 'Last Row
82         If LastRow < 4 Then Exit Sub
83         .Range("A2:K" & LastRow).AdvancedFilter xlFilterCopy, CriteriaRange:=.Range(
84             "L2:M3" ), CopyToRange:=.Range("P2:T2" ), Unique:=True
85         LastResultRow = .Range("P99999" ).End(xlUp).Row
86         If LastResultRow < 3 Then Exit Sub
87         Payments.Range("B2" ).Value = True 'Set Payment Load to true
88         .Range("S3:S" & LastResultRow).Formula = .Range("S1" ).Formula 'Total Payments
89             Formula
90         Payments.Range("E11:I" & LastResultRow + 8).Value = .Range("P3:T" &
91             LastResultRow).Value
92     End With
93     Payments.Range("B2" ).Value = False 'Set Payment Load to False
94 End Sub
95
96 Sub Payments_Next()
97     Dim MaxPayID As Long, PayID As Long
98     With Payments
99         On Error Resume Next
100        MaxPayID = Application.WorksheetFunction.Max(PaymentList.Range("Pay_ID" ))
101        On Error GoTo 0
102        If MaxPayID = 0 Then
103            MsgBox "Please create a payment first"
104            Exit Sub
105        End If
106        PayID = .Range("B3" ).Value 'Payment ID
107
108        If PayID = 0 Or .Range("B4" ).Value = Empty Then 'Load Last Payment Created
109            PayRow = 4 'On new Payment, Go first one created
110        Else
111            PayRow = .Range("B4" ).Value + 1 'Pay Row + 1
112        End If

```

```

107
108
109     If MaxPayID = PayID Then 'Last Payment
110         MsgBox "You are on the last payment"
111         Exit Sub
112     End If
113
114     .Range("B3" ).Value = PaymentList.Range("A" & PayRow).Value 'Set Payment ID
115     Payments_Load 'Load Payment
116 End With
117 End Sub
118
119 Sub Payments_Previous()
120     Dim MinPayID As Long, PayID As Long
121     With Payments
122         On Error Resume Next
123         MinPayID = Application.WorksheetFunction.Min(PaymentList.Range("Pay_ID" ))
124         On Error GoTo 0
125         If MinPayID = 0 Then
126             MsgBox "Please create a payment first"
127             Exit Sub
128         End If
129         PayID = .Range("B3" ).Value 'Payment ID
130         If PayID = 0 Or .Range("B4" ).Value = Empty Then 'Load Last Payment Created
131             PayRow = PaymentList.Range("A9999" ).End(xlUp).Row 'Last Row
132         Else
133             PayRow = .Range("B4" ).Value - 1 'Pay Row -1
134         End If
135
136         If PayRow = 3 Or MinPayID = .Range("B3" ).Value Then 'First Payment
137             MsgBox "You are on the first payment"
138             Exit Sub
139         End If
140
141         .Range("B3" ).Value = PaymentList.Range("A" & PayRow).Value 'Set Payment ID
142         Payments_Load 'Load Payment
143     End With
144 End Sub
145
146 Sub Payments_SaveUpdate()
147     With Payments
148         'Check For Required Fields
149         If .Range("F3" ).Value = Empty Or .Range("J3" ).Value = Empty Or .Range("J5" ).
150             Value = Empty Then
151             MsgBox "Please make sure to add in a Customer, Payment Date and Payment
152                 Amount before saving"
153             Exit Sub
154         End If
155
156         If .Range("J5" ).Value > .Range("J9" ).Value Then
157             If MsgBox("The Payment Amount is greater than then Applied Amount. Would you
158                 like to create Customer Credit?" , vbYesNo, "Overpayment?" ) = vbNo Then Exit
159             Sub
160                 Customer_CreateCredit
161                 Payments_LoadOpenInvoices
162             End Sub
163         End If
164     End With
165 End Sub

```

```

1 2
163 'Check To Make sure Payment Amount = Applied amount
164 If .Range("J5" ).Value <> .Range("J9" ).Value Then
165     MsgBox "Please make sure Payment Amount is equal to Applied Amount"
166     Exit Sub
167 End If
168
169 If .Range("B4" ).Value = Empty Then 'New Payment
170     PayRow = PaymentList.Range("A99999" ).End(xlUp).Row + 1 'First Avail Row
171     .Range("B3" ).Value = .Range("B5" ).Value 'Next payment ID
172     PaymentList.Range("A" & PayRow).Value = .Range("B3" ).Value 'Pay ID
173 Else 'Existing Payment
174     PayRow = .Range("B4" ).Value
175 End If
176 For PayCol = 2 To 6
177     PaymentList.Cells(PayRow, PayCol).Value = .Range(PaymentList.Cells(1, PayCol).
178     Value).Value 'Save Form Data
179 Next PayCol
180
181 'Save Pay Items
182 LastPayItemRow = .Range("E99999" ).End(xlUp).Row 'Last Pay Item
183 For PayItemRow = 11 To LastPayItemRow
184     If .Range("D" & PayItemRow).Value = Chr(252) Then 'Selected Row
185         If .Range("K" & PayItemRow).Value = Empty Then 'New Pay Item Row
186             PayItemDBRow = PayItems.Range("A99999" ).End(xlUp).Row + 1 'First Avail
187             Pay Item Row
188             PayItems.Range("A" & PayItemDBRow).Value = .Range("B3" ).Value
189             'Payment ID
190             PayItems.Range("F" & PayItemDBRow).Value = "=Row()" 'Item Row Formula
191             .Range("K" & PayItemRow).Value = PayItemDBRow 'Database Row
192         Else 'Previously Saved
193             PayItemDBRow = .Range("K" & PayItemRow).Value 'Existing Pay Item Row
194         End If
195         PayItems.Range("B" & PayItemDBRow).Value = .Range("F" & PayItemRow).Value
196         'Invoice ID
197         PayItems.Range("C" & PayItemDBRow).Value = .Range("F3" ).Value 'Customer
198         PayItems.Range("D" & PayItemDBRow).Value = .Range("J3" ).Value 'Pay Date
199         PayItems.Range("E" & PayItemDBRow).Value = .Range("J" & PayItemRow).Value
200         'Amount
201     End If
202 Next PayItemRow
203 End With
204 End Sub

```

```

1 Dim SelCell As Range
2 Dim DayName As String
3
4
5
6 .....''Color Calendar Background''.....
7 Sub CalCol()
8     With ActiveSheet.Shapes.Range(Array("CalBack" , "Settings" )).Select
9         With Selection.ShapeRange.Fill
10            .ForeColor.RGB = ActiveSheet.Shapes(Application.Caller).Fill.ForeColor.RGB
11        End With
12        ActiveSheet.Range(Sheets("CalPopUp" ).Range("A7" ).Value).Select
13    End With
14 End Sub
15 Sub CalendarHide()
16     Dim DayNum As Long
17     'Hide Calendar, Reset Day Colors
18     On Error GoTo NoCal
19     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
20     Sheets("CalPopUp" ).Range("A7" ).Value = ""
21     If Sheets("CalPopUp" ).Range("A20" ).Value <> Empty Then
22         For DayNum = 1 To 42
23             DayName = DayNum & "Day"
24             With ActiveSheet.Shapes(DayName)
25                 .Fill.ForeColor.RGB = RGB(255, 255, 255)
26                 .TextFrame2.TextRange.Font.Bold = msoFalse
27             End With
28         Next DayNum
29     End If
30     Exit Sub
31     NoCal: 'If calendar has been removed by accident, paste in backup calendar from
           CalPopUp Sheet
32 End Sub
33
34 Sub CalendarShow()
35     With ActiveSheet
36         Set SelCell = ActiveCell
37         'Check if active cell is a valid date
38         If IsDate(SelCell.Value) = True Then
39             Sheets("CalPopUp" ).Range("A1" ).Value = SelCell.Value
40         Else: 'If No Date or incorrect Date user current date
41             Sheets("CalPopUp" ).Range("A1" ).Value = "=Today()"
42         End If
43         'Clear all shapes to white (if calendar is visible)
44         If ActiveSheet.Shapes("Calendar" ).Visible = True Then
45             For DayNum = 1 To 42
46                 DayName = DayNum & "Day"
47                 With ActiveSheet.Shapes(DayName)
48                     .Fill.ForeColor.RGB = RGB(255, 255, 255)
49                     .TextFrame2.TextRange.Font.Bold = msoFalse
50                 End With
51             Next DayNum
52         End If
53
54         Sheets("CalPopUp" ).Range("A3" ).Value = Month(Sheets("CalPopUp" ).Range("A1" ).
           Value) 'Set Month
55         Sheets("CalPopUp" ).Range("A2" ).Value = Year(Sheets("CalPopUp" ).Range("A1" ).
           Value) 'Set Year
56         DayName = Sheets("CalPopUp" ).Range("A20" ).Value & "Day"
57         ' UnGroupCal

```

1 2

```

1 2
58 If InStr(.Shapes("1Day").DrawingObject.Formula, "J" ) <> 0 Then 'Run Workbook
    Link Remover and Cell Link Replacement
59     MacroLinkRemover
60     CalFormulaReplacement
61 End If
62 'GroupCal
63 On Error GoTo NoCal
64 If DayName = "Day" Then DayName = Day(Date) & "Day" 'Set Default Day
65 With ActiveSheet.Shapes(DayName)
66     .Fill.ForeColor.RGB = RGB(252, 213, 180)
67     .TextFrame2.TextRange.Font.Bold = msoTrue
68 End With
69 On Error GoTo NoCal
70 .Shapes("Calendar").Visible = msoCTrue
71 .Shapes.Range(Array("Settings", "CalCol1", "CalCol2", "CalCol3", "CalCol4",
    "CalCol5", "CalCol6", "CalCol7", "CalCol8", "CalCol9")).Visible = False '
72 .Shapes("Calendar").Left = SelCell.Left
73 .Shapes("Calendar").Placement = xlMove
74 .Shapes("Calendar").Top = SelCell.Offset(1, 0).Top
75 If Sheets("CalPopUp").Range("A6").Value > 0 Then
76     .Shapes.Range(Array("36Day", "37Day", "38Day", "39Day", "40Day", "41Day",
    "42Day")).Visible = True
77 Else:
78     .Shapes.Range(Array("36Day", "37Day", "38Day", "39Day", "40Day", "41Day",
    "42Day")).Visible = False
79 End If
80 Sheets("CalPopUp").Range("A7").Value = SelCell.Address
81 ActiveCell.Select
82 End With
83 Exit Sub
84 NoCal:
85 MsgBox "The Pop-up Calendar does not exist on this worksheet. Please copy the
    calendar over from another sheet and paste into this sheet"
86 End Sub
87 Sub CalFormulaReplacement()
88 With ActiveSheet
89     Dim DayNum, ColNum, RowNum As Long
90     Dim Shp As Shape
91     ColNum = 2
92     RowNum = 1
93     For DayNum = 1 To 42
94         .Shapes(DayNum & "Day").DrawingObject.Formula = "=CalPopUp!" & .Cells(RowNum
    , ColNum).Address
95         ColNum = ColNum + 1
96         If ColNum = 9 Then
97             ColNum = 2
98             RowNum = RowNum + 1
99         End If
100     Next DayNum
101     .Shapes("Month").DrawingObject.Formula = "=CalPopUp!$A$4"
102     .Shapes("Year").DrawingObject.Formula = "=CalPopUp!$A$2"
103 End With
104 End Sub
105
106 Sub CheckForSheet()
107 'Checks for existance of Calendar Pop-up Worksheet
108 Dim ws As Worksheet
109 On Error GoTo CreateWS
110 Set ws = ActiveWorkbook.Sheets("CalPopUp")
111 Exit Sub

```

```

1
112 CreateWS:
113 CreateCalSht
114 End Sub
115
116
117 'Create Calendar Sheet on First Run of Calendar
118 Sub CreateCalSht()
119 Dim ColCnt, RowCnt, DayCnt, CalCol As Long
120 Dim ws, ActSht As Worksheet
121 Set ActSht = ActiveSheet
122 'On Error GoTo NoCal
123 ActiveSheet.Shapes("Calendar").Copy
124 Set ws = ThisWorkbook.Sheets.Add(After:=ThisWorkbook.Sheets(ThisWorkbook.Sheets.
Count))
125 ws.Name = "CalPopUp"
126 ActSht.Activate
127
128 'Reassign Shape Links & Macros
129 With ActiveSheet
130 UnGroupCal
131 Unprotect
132 .Shapes("PrevYr").OnAction = "" & ActiveWorkbook.Name & "!PrevYear"
133 .Shapes("NextYr").OnAction = "" & ActiveWorkbook.Name & "!NextYear"
134 .Shapes("NextRec").OnAction = "" & ActiveWorkbook.Name & "!NextMonth"
135 .Shapes("NextTri").OnAction = "" & ActiveWorkbook.Name & "!NextMonth"
136 .Shapes("PrevRec").OnAction = "" & ActiveWorkbook.Name & "!PrevMonth"
137 .Shapes("PrevTri").OnAction = "" & ActiveWorkbook.Name & "!PrevMonth"
138 .Shapes("SetBtn").OnAction = "" & ActiveWorkbook.Name & "!ShowSettings"
139 .Shapes("Month").DrawingObject.Formula = "=CalPopUp!A4"
140 .Shapes("Year").DrawingObject.Formula = "=CalPopUp!A2"
141 DayCnt = 1
142 For RowCnt = 1 To 6
143 For ColCnt = 2 To 8
144 .Shapes(DayCnt & "Day").DrawingObject.Formula = "=CalPopUp!" & .Cells(
RowCnt, ColCnt).Address 'Assigned Linked Cell
145 .Shapes(DayCnt & "Day").OnAction = "" & ActiveWorkbook.Name &
"!SelectDay" 'Assign Macro
146 DayCnt = DayCnt + 1
147 Next ColCnt
148 Next RowCnt
149
150 'Assign Color Macros
151 For CalCol = 1 To 9
152 .Shapes("CalCol" & CalCol).OnAction = "" & ActiveWorkbook.Name &
"!CalCol" 'Assign Color Macro
153 Next CalCol
154 End With
155
156 With Sheets("CalPopUp")
157 .Paste
158 .Visible = xlSheetHidden
159
160 'Add in Formulas and Details
161 .Range("A1").Value = Date 'Set Current Date
162 .Range("A2").Value = Year(Date) 'Set Current Year
163 .Range("A3").Value = Month(Date) 'Set Current Month #
164 .Range("A4").Value = "=INDEX(CalMonths,A3,)"
165 .Range("A5").Value = "=A4&" & Chr(34) & " " & Chr(34) & "&CalYear"
166 .Range("A6").Value = "=SUM(B6:H6)"
167 .Range("A8").Value = "January"

```

1 2

```

1 2
168 .Range("A8" ).AutoFill Destination:=.Range("A8:A19" ), Type:=xlFillDefault
169 .Range("A20" ).Value =
    "=IFERROR(INDIRECT(ADDRESS(SUMPRODUCT((B1:H6=A1)*ROW(B1:H6))+6,SUMPRODUCT((B1:H6=A
1)*COLUMN(B1:H6)),1,1))," & Chr(34) & Chr(34) & ")"
170 'Set Defined Names
171 ActiveWorkbook.Names.Add Name:="CalMonths" , RefersTo:="=CalPopUp!$A$8:$A$19"
172 ActiveWorkbook.Names.Add Name:="CalYear" , RefersTo:="=CalPopUp!$A$2"
173
174 'Add in Calendar Formulas
175
176 .Range("B1" ).Value =
    "=IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=1,DATE(CalYear,MATCH($A$4,Cal
1Months,0),1)," & Chr(34) & Chr(34) & ")"
177 .Range("C1" ).Value = "=IF(B1<>" & Chr(34) & Chr(34) &
    ",B1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=2,DATE(CalYear,MATCH($A
1$4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")"
178 .Range("D1" ).Value = "=IF(C1<>" & Chr(34) & Chr(34) &
    ",C1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=3,DATE(CalYear,MATCH($A
1$4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")"
179 .Range("E1" ).Value = "=IF(D1<>" & Chr(34) & Chr(34) &
    ",D1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=4,DATE(CalYear,MATCH($A
1$4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")"
180 .Range("F1" ).Value = "=IF(E1<>" & Chr(34) & Chr(34) &
    ",E1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=5,DATE(CalYear,MATCH($A
1$4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")"
181 .Range("G1" ).Value = "=IF(F1<>" & Chr(34) & Chr(34) &
    ",F1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=6,DATE(CalYear,MATCH($A
1$4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")"
182 .Range("H1" ).Value = "=IF(G1<>" & Chr(34) & Chr(34) &
    ",G1+1,IF(WEEKDAY(DATE(CalYear,MATCH($A$4,CalMonths,0),1))=7,DATE(CalYear,MATCH($A
1$4,CalMonths,0),1)," & Chr(34) & Chr(34) & ")"
183 .Range("B2" ).Value = "=H1+1"
184 .Range("C2" ).Value = "=B2+1"
185 .Range("C2" ).AutoFill Destination:=.Range("C2:H2" ), Type:=xlFillDefault
186 .Range("B2:H2" ).AutoFill Destination:=.Range("B2:H4" ), Type:=xlFillDefault
187 .Range("B5" ).Value = "=IF(OR(H4=" & Chr(34) & Chr(34) & ",MONTH(H4+1)<>$A$3),"
    & Chr(34) & Chr(34) & ",H4+1)"
188 .Range("C5" ).Value = "=IFERROR(IF(MONTH(B5+1)<>$A$3," & Chr(34) & Chr(34) &
    ",B5+1)," & Chr(34) & Chr(34) & ")"
189 .Range("B6" ).Value = "=IFERROR(IF(OR(H5=" & Chr(34) & Chr(34) &
    ",MONTH(H5+1)<>$A$3)," & Chr(34) & Chr(34) & ",H5+1)," & Chr(34) & Chr(34) &
    ")")
190 .Range("C6" ).Value = "=IFERROR(F(MONTH(I5+1)<>$A$3," & Chr(34) & Chr(34) &
    ",I5+1)," & Chr(34) & Chr(34) & ")"
191 .Range("C5:C6" ).AutoFill Destination:=.Range("C5:H6" ), Type:=xlFillDefault
192
193 'Set format to Single Day
194 .Range("B1:H6" ).NumberFormat = "d"
195
196 'Add in relative Day #'s
197 .Range("B7" ).Value = "1"
198 .Range("C7" ).Value = "2"
199 .Range("B8" ).Value = "8"
200 .Range("C8" ).Value = "9"
201 .Range("B7:C8" ).AutoFill Destination:=.Range("B7:H8" ), Type:=xlFillDefault
202 .Range("B7:H8" ).AutoFill Destination:=.Range("B7:H12" ), Type:=xlFillDefault
203 GroupCal
204 End With
205 Exit Sub
206 NoCal:

```

```

1
207 MsgBox "The Pop-up Calendar does not exist on this worksheet. Please copy the
calendar over from another sheet and paste into this sheet"
208 End Sub
209 Sub GroupCal()
210 ActiveSheet.Shapes.Range(Array("NextTri" , "NextRec" )).Group.Select
211 Selection.ShapeRange.Name = "NextMonth"
212 ActiveSheet.Shapes.Range(Array("PrevTri" , "PrevRec" )).Group.Select
213 Selection.ShapeRange.Name = "PrevMonth"
214 ActiveSheet.Shapes.Range(Array("Settings" , "40Day" , "41Day" , "39Day" , "38Day" _
215 , "42Day" , "37Day" , "36Day" , "CalBack" , "Month" , "Year" , "CalBorder" ,
"1Day" , _
216 "3Day" , "14Day" , "7Day" , "4Day" , "2Day" , "5Day" , "8Day" , "10Day" ,
"6Day" , _
217 "13Day" , "11Day" , "9Day" , "12Day" , "15Day" , "17Day" , "20Day" , "21Day" ,
"18Day" _
218 , "16Day" , "19Day" , "22Day" , "24Day" , "26Day" , "27Day" , "25Day" , "23Day"
, _
219 "28Day" , "29Day" , "31Day" , "34Day" , "35Day" , "32Day" , "30Day" , "33Day" ,
"Sa" , _
220 "Fr" , "Th" , "We" , "Tu" , "Mo" , "Su" , "SetBtn" , "CalCol1" , "CalCol2" ,
"CalCol3" , _
221 "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" ,
"PrevMonth" , _
222 "NextMonth" , "NextYr" , "PrevYr" ).Visible = msoCTrue
223 ActiveSheet.Shapes.Range(Array("Settings" , "40Day" , "41Day" , "39Day" , "38Day" _
224 , "42Day" , "37Day" , "36Day" , "CalBack" , "Month" , "Year" , "CalBorder" , "1Day"
, _
225 "3Day" , "14Day" , "7Day" , "4Day" , "2Day" , "5Day" , "8Day" , "10Day" , "6Day" , _
226 "13Day" , "11Day" , "9Day" , "12Day" , "15Day" , "17Day" , "20Day" , "21Day" ,
"18Day" _
227 , "16Day" , "19Day" , "22Day" , "24Day" , "26Day" , "27Day" , "25Day" , "23Day" , _
228 "28Day" , "29Day" , "31Day" , "34Day" , "35Day" , "32Day" , "30Day" , "33Day" ,
"Sa" , _
229 "Fr" , "Th" , "We" , "Tu" , "Mo" , "Su" , "SetBtn" , "CalCol1" , "CalCol2" ,
"CalCol3" , _
230 "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" , "PrevMonth"
, _
231 "NextMonth" , "NextYr" , "PrevYr" )).Select
232 Selection.ShapeRange.Group.Select
233 Selection.ShapeRange.Name = "Calendar"
234 Selection.Name = "Calendar"
235 Selection.Placement = xlMove
236 ActiveSheet.Shapes("Calendar" ).Placement = 2
237 End Sub
238
239
240
241
242 Sub MacroLinkRemover()
243 'PURPOSE: Remove an external workbook reference from all shapes triggering macros
244 'Source: www.ExcelForFreelancers.com
245 Dim Shp As Shape
246 Dim MacroLink, NewLink As String
247 Dim SplitLink As Variant
248
249 For Each Shp In ActiveSheet.Shapes 'Loop through each shape in worksheet
250
1 2

```

```

1 2
251 'Grab current macro link (if available)
252 On Error GoTo NextShp
253 MacroLink = Shp.OnAction
254
255 'Determine if shape was linking to a macro
256 If MacroLink <> "" And InStr(MacroLink, "!") <> 0 Then
257     'Split Macro Link at the exclamation mark (store in Array)
258     SplitLink = Split(MacroLink, "!")
259
260     'Pull text occurring after exclamation mark
261     NewLink = SplitLink(1)
262
263     'Remove any straggling apostrophes from workbook name
264     If Right(NewLink, 1) = "'" Then
265         NewLink = Left(NewLink, Len(NewLink) - 1)
266     End If
267
268     'Apply New Link
269     Shp.OnAction = NewLink
270 End If
271 NextShp:
272 Next Shp
273 End Sub
274 Sub NextMonth()
275     'Next Month button
276     If Sheets("CalPopUp").Range("A20").Value <> Empty Then
277         DayName = Sheets("CalPopUp").Range("A20").Value & "Day"
278         With ActiveSheet.Shapes(DayName)
279             .Fill.ForeColor.RGB = RGB(255, 255, 255)
280             .TextFrame2.TextRange.Font.Bold = msoFalse
281         End With
282     End If
283     With Sheets("CalPopUp")
284         If .Range("A3").Value = 12 Then
285             .Range("A3").Value = 1
286             .Range("A2").Value = .Range("A2").Value + 1
287         Else:
288             .Range("A3").Value = .Range("A3").Value + 1
289         End If
290         If .Range("A6").Value > 0 Then
291             ActiveSheet.Shapes.Range(Array("36Day", "37Day", "38Day", "39Day",
292             "40Day", "41Day", "42Day")).Visible = True
293         Else:
294             ActiveSheet.Shapes.Range(Array("36Day", "37Day", "38Day", "39Day",
295             "40Day", "41Day", "42Day")).Visible = False
296         End If
297     End With
298 End Sub
299 Sub NextYear()
300     ThisWorkbook.Sheets("CalPopUp").Range("A2").Value = ThisWorkbook.Sheets(
301     "CalPopUp").Range("A2").Value + 1
302 End Sub
303 Sub PrevMonth()
304     'Previous Month Button
305     If Sheets("CalPopUp").Range("A20").Value <> Empty Then
306         DayName = Sheets("CalPopUp").Range("A20").Value & "Day"
307         With ActiveSheet.Shapes(DayName)
308             .Fill.ForeColor.RGB = RGB(255, 255, 255)
309             .TextFrame2.TextRange.Font.Bold = msoFalse
310         End With
311     End If
312 End Sub
1 2 3

```

```

1 2 3
308     End With
309     End If
310     With Sheets("CalPopUp" )
311         If .Range("A3" ).Value = 1 Then
312             .Range("A3" ).Value = 12
313             .Range("A2" ).Value = .Range("A2" ).Value - 1
314         Else:
315             .Range("A3" ).Value = .Range("A3" ).Value - 1
316         End If
317         If .Range("A6" ).Value > 0 Then
318             ActiveSheet.Shapes.Range(Array("36Day" , "37Day" , "38Day" , "39Day" ,
319             "40Day" , "41Day" , "42Day" )).Visible = True
320         Else:
321             ActiveSheet.Shapes.Range(Array("36Day" , "37Day" , "38Day" , "39Day" ,
322             "40Day" , "41Day" , "42Day" )).Visible = False
323         End If
324     End With
325 End Sub
326
327 Sub PrevYear()
328     ThisWorkbook.Sheets("CalPopUp" ).Range("A2" ).Value = ThisWorkbook.Sheets(
329     "CalPopUp" ).Range("A2" ).Value - 1
330 End Sub
331
332 Sub ReplaceCalendar() 'Shape Deleted
333     ThisWorkbook.Sheets("CalPopUp" ).Shapes("Calendar" ).Copy 'Copy From Developers
334     sheet
335     ActiveCell.Select 'Select the active Cell
336     ActiveSheet.Paste 'Paste in Calendar
337 End Sub
338
339 .....
340
341 ''''Select Day Of The Month
342 .....
343
344 Sub SelectDay()
345     Dim DayNumb As Long, RowNumb As Long, ColNumb As Long
346     DayNumb = Replace(Application.Caller, "Day" , "" )
347     RowNumb = Application.WorksheetFunction.RoundUp(DayNumb / 7, 0)
348     ColNumb = DayNumb Mod 7 + 1
349     If ColNumb = 1 Then ColNumb = 8
350     'On Error Resume Next
351     If ThisWorkbook.Sheets("CalPopUp" ).Range("A7" ).Value = Empty Then Exit Sub
352     ActiveSheet.Range(ThisWorkbook.Sheets("CalPopUp" ).Range("A7" ).Value).Value =
353     ThisWorkbook.Sheets("CalPopUp" ).Cells(RowNumb, ColNumb).Value
354     ActiveSheet.Shapes("Calendar" ).Visible = msoFalse
355     ActiveCell.Offset(0, 1).Select
356 End Sub
357
358 Sub ShowSettings()
359     'Show or Hide Calendar Settings Panel
360     If ActiveSheet.Shapes.Range(Array("Settings" )).Visible = True Then
361         ActiveSheet.Shapes.Range(Array("Settings" , "CalCol1" , "CalCol2" , "CalCol3" ,
362         "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" )).Visible
363         = False
364     Else:
365         ActiveSheet.Shapes.Range(Array("Settings" , "CalCol1" , "CalCol2" , "CalCol3" ,
366         "CalCol4" , "CalCol5" , "CalCol6" , "CalCol7" , "CalCol8" , "CalCol9" )).Visible
367         = True
368     End If
369 End Sub
370
371 Sub UngroupCal()
372

```

```
1
359 On Error Resume Next
360 ActiveSheet.Shapes("Calendar").Ungroup
361 ActiveSheet.Shapes("NextMonth").Ungroup
362 ActiveSheet.Shapes("PrevMonth").Ungroup
363 On Error GoTo 0
364 End Sub
```

```
1 Option Explicit
2 Dim LastPage As Long, PageNumb As Long
3 Sub Invoice_Print()
4     With Invoice
5         Invoice_SaveUpdate 'Save Current Invoice
6         LastPage = .Range("B12" ).Value 'Set Last Page
7         For PageNumb = 1 To LastPage
8             .Range("B11" ).Value = PageNumb
9             Invoice_PageLoad 'Load Page
10            .PrintOut , , , False, True, , , , False
11        Next PageNumb
12    End With
13 End Sub
14
15 Sub Test()
16     Application.CellDragAndDrop = True
17 End Sub
18
```

—, 7, 15, 21, 35

A

Activate, 33
 ActiveCell, 31, 32, 37
 ActiveSheet, 23, 31-33, 35-38
 ActiveWorkbook, 32-34
 ActSht, 33
 Add, 19, 22, 27, 33, 34
 AddCustForm, 18, 22
 Address, 8, 32, 33
 Admin, 6, 8, 19, 23
 AdvancedFilter, 22, 24, 27, 28
 After, 33
 AllowMultiSelect, 19
 AppEvents_Start, 19, 22, 23, 26
 AppEvents_Stop, 19, 22, 23, 25
 Application, 5, 18, 19, 24-26, 28, 29, 31, 37, 39
 Apply, 22, 27
 AppSumm, 21
 Array, 9, 31, 32, 35-37
 AutoFill, 34

B

body, 21
 Bold, 31, 32, 36
 BtnMov, 19

C

Cal, 6
 CalCol, 31, 33
 Calculation, 19
 CalendarHide, 7, 31
 CalendarShow, 7, 31
 CalFormulaReplacement, 32
 Caller, 26, 31, 37
 CancelBtn_Click, 18
 CellDragAndDrop, 39
 Cells, 6-8, 18, 22, 23, 26, 27, 30, 32, 33, 37
 CheckForLbl, 8
 CheckForSheet, 7, 32
 Chr, 3, 7, 15, 21, 23, 30, 33, 34
 Clear, 22, 27
 ClearContents, 3, 5, 7, 15, 23, 24, 27, 28
 ColCnt, 33
 ColNum, 32
 ColNumb, 37
 Column, 6
 Columns, 7, 8
 ColumnWidth, 7, 8
 Company_AddLogo, 19
 Control, 18, 22
 Controls, 18, 22
 Copy, 33, 37
 CopyToRange, 22, 24, 27, 28
 Count, 33
 CountLarge, 3, 5, 6, 15
 CrAmnt, 20
 CreateCalSht, 33
 CreateItem, 21
 CreateObject, 21
 CreateWS, 32, 33
 CriteriaRange, 22, 24, 27, 28
 CustCol, 18, 22
 CustEmail, 21
 CustNm, 20, 21
 Customer_AddUpdate, 22

Customer_CreateCredit, 20, 29
 CustomerFld, 18, 22
 Customers, 18, 21, 22
 CustRow, 18, 21, 22

D

DataOption, 22, 27
 Day, 32
 DayCnt, 33
 DayName, 31, 32, 36
 DayNum, 31, 32
 DayNumb, 37
 Delay, 25
 Delete, 22, 27
 DescWidth, 7, 8
 Destination, 34
 Dir, 8
 display, 21
 DoEvents, 25
 DrawingObject, 8, 32, 33
 Duplicate, 8

E

EditMode, 19
 Email_Summary, 21
 Empty, 3, 5-9, 15, 18, 21-25, 27-31, 36, 37
 EnableEvents, 5, 19
 EntireColumn, 7
 EntireRow, 22, 27
 Explicit, 3-5, 10-22, 27, 39

F

Field1, 18
 FieldRow, 22, 23, 26
 FileDialog, 19
 Fill, 19, 25, 31, 32, 36
 Filters, 19
 Find, 5, 6, 21, 23-25
 Font, 31, 32, 36
 ForeColor, 19, 31, 32, 36
 Format, 21
 Formula, 8, 28, 32, 33

G

Group, 9, 35
 GroupCal, 34, 35
 GrpShp, 6, 8

H

HeadShp, 22
 Hidden, 7
 Hide, 18

I

i, 25
 Insert, 8
 InStr, 21, 32, 36
 Intersect, 3, 5-7, 15
 InvItemRow, 5, 6, 22, 24
 InvNumb, 22, 24, 25
 Invoice, 18-20, 22-25, 39
 Invoice_Delete, 22
 Invoice_Load, 5, 23-25
 Invoice_New, 20, 22, 23
 Invoice_NewPage, 23
 Invoice_NextInvoice, 23
 Invoice_NextPage, 24
 Invoice_PageLoad, 23-25, 39
 Invoice_PrevInvoice, 24
 Invoice_PrevPage, 25

Invoice_Print, 20, 39
 Invoice_SavedMsg, 25, 26
 Invoice_SaveUpdate, 20, 23, 25, 39
 InvoiceItems, 6, 22, 24
 InvoiceList, 22-26, 28
 InvRow, 21-26
 InvWidth, 7, 8
 IsDate, 31
 ItemCol, 5-8
 ItemDBRow, 5, 22
 ItemLblShp, 6-9
 ItemNm, 6, 8, 9
 ItemRow, 5, 7, 8, 22
 Items, 5
 ItemShp, 6-9
 ItemTypeRow, 5-8

K

Key, 22, 27

L

LastItemRow, 22
 LastPage, 39
 LastPayItemRow, 27, 30
 LastResultRow, 22, 24, 27, 28
 LastRow, 21, 22, 24, 27, 28
 Left, 8, 19, 32, 36
 Len, 36
 LogoPic, 19

M

MacroLink, 35, 36
 MacroLinkRemover, 32, 35
 Max, 18, 24, 28
 MaxInvNumb, 23, 24
 MaxPayID, 28, 29
 Msg, 21
 Min, 25, 29
 MinInvNumb, 24, 25
 MinPayID, 29
 Month, 31, 33
 MovInc, 19
 MsgBox, 5, 18, 21-25, 27-30, 32, 35
 msoCTrue, 6, 9, 25, 32, 35
 msoFalse, 6, 9, 25, 31, 36, 37
 msoFileDialogFilePicker, 19
 msoTrue, 32

N

Name, 8, 9, 23, 33-35
 Names, 34
 NewLink, 35, 36
 NextMonth, 36
 NextShp, 36
 NextYear, 36
 NoCal, 31, 32, 34
 NoData, 28
 NolItems, 24
 NoSelection, 19
 NotSaved, 22, 27
 NumberFormat, 34

O

Offset, 9, 32, 37
 OnAction, 33, 36
 Order, 22, 27
 OutApp, 21
 OutMail, 21

P

PageNumb, 39
 Paste, 33, 37
 PayAmt, 21
 PayCol, 27, 28, 30
 PayDate, 21
 PayID, 28, 29
 PayItemDBRow, 27, 30
 PayItemRow, 27, 30
 PayItems, 27, 28, 30
 PaymentList, 27-30
 Payments, 20, 21, 27-29
 Payments_AddNew, 27
 Payments_Delete, 27
 Payments_Load, 27, 29
 Payments_LoadOpenInvoices, 15, 28, 29
 Payments_Next, 28
 Payments_Previous, 29
 Payments_SaveUpdate, 29
 PayRow, 27-30
 PicPath, 19
 Pictures, 8
 Placement, 8, 9, 32, 35
 PrevMonth, 36
 PrevYear, 37
 PrintOut, 39

R

Range, 3, 5-9, 15, 18-37, 39
 RefersTo, 34
 Replace, 21, 37
 ReplaceCalendar, 6, 37
 ResultRow, 22, 24, 27
 RGB, 19, 31, 32, 36
 Right, 36
 RoundUp, 37
 Row, 3, 5-8, 15, 18, 21-25, 27-30
 RowCnt, 33
 RowNum, 32
 RowNumb, 37

S

SaveBtn_Click, 18
 ScreenUpdating, 19
 SelCell, 31, 32
 SelectDay, 37
 SelectedItems, 19
 Selection, 31, 35
 SelRow, 5, 6
 SetRange, 22, 27
 Shape, 6, 22, 32, 35
 ShapeRange, 31, 35
 Shapes, 6-9, 19, 20, 25, 31-33, 35-38
 Sheets, 31-33, 36, 37
 Show, 19, 22
 ShowSettings, 37
 Shp, 32, 35, 36
 SingleRow, 22
 SkipSort, 27
 Sort, 22, 27
 SortFields, 22, 17
 SortOn, 22, 27
 Split, 36
 SplitLink, 35, 36
 StartTime, 25
 StatusRow, 22, 23
 Subj, 21
 Subject, 21

T

Target, 3, 5-9, 15
 TermRow, 5, 6, 22, 23
 Test, 39
 Text, 19, 20
 TextFrame2, 19, 20, 31, 32, 36
 TextRange, 19, 20, 31, 32, 36
 ThisWorkbook, 33, 36, 37
 Timer, 25
 Title, 19
 Top, 8, 32
 Transparency, 25

U

Ungroup, 9, 38
 UnGroupCal, 33, 37
 Unique, 22, 24, 27, 28
 Unload, 22

V

Value, 3, 5-9, 15, 18-34, 36, 37, 39
 vbCrLf, 21
 vbDirectory, 8
 vbNo, 22, 27, 29
 vbYesNo, 22, 27, 29
 Visible, 6, 7, 9, 25, 31-33, 35-37

W

Worksheet, 32, 33
 Worksheet_Change, 5, 15
 Worksheet_SelectionChange, 3, 6, 15
 WorksheetFunction, 18, 24, 25, 28, 29, 37
 ws, 32, 33

X

xlCalculationAutomatic, 19
 xlCalculationManual, 19
 xlDescending, 22, 27
 xlFillDefault, 34
 xlFilterCopy, 22, 24, 27, 28
 xlMove, 8, 9, 32, 35
 xlSheetHidden, 33
 xlSortNormal, 22, 27
 xlSortOnValues, 22, 27
 xlUp, 6, 18, 21, 22, 24, 25, 27-30
 xlValues, 5, 6, 21, 23-25
 xlWhole, 5, 6, 21, 23-25

Y

Year, 31, 33

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