# Creating a Cell Formatter Macro with Pop-up

Author: Muhammad Arif Bin Rawi

Date: 12 April 2021

Preface

This guide assumes that you have already added the "developer" tab to your Excel tabs and that you have a basic understanding of Excel VBA. Ensure that you save the files as "macro-enabled" format (.xlsm). Also, as you will be working with VBA within work environment, you will need to know how to self-cert your macro.
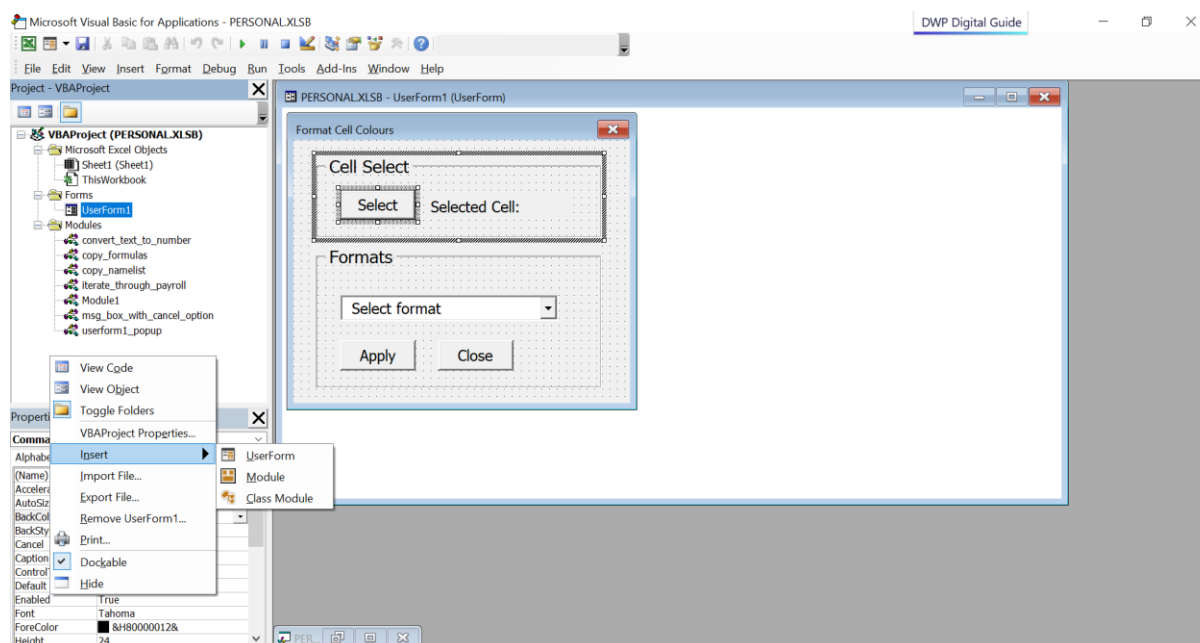
The Interface

Interface Layout
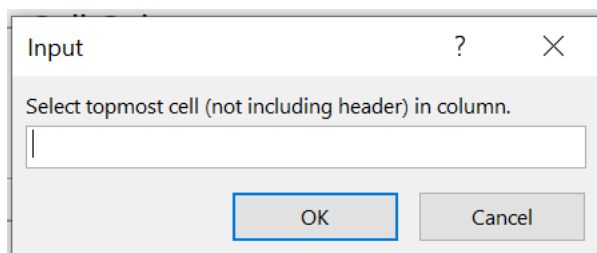
The interface is created using UserForm and is meant to be intuitive. At one glance, users should understand that they first have to select the cell they wish to start from. Then from there, select what kind of formats they wish to apply.
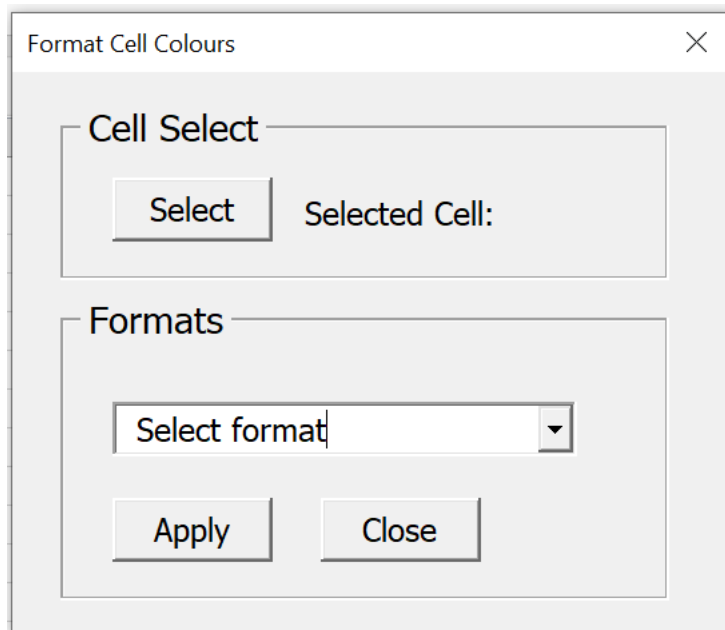
Clicking on the "Select" button will display a prompt that has a short instruction.

Clicking on the dropdown will show the formats available to apply.
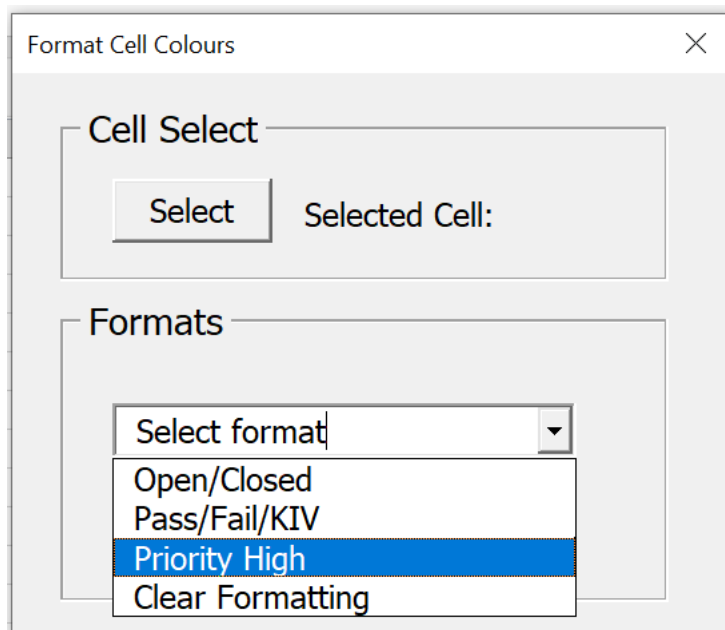
Editing the interface

The font, size and label of the interface can be changed. For example, the "Select" button interface.

Click on the button and look at the properties window.

Properties - CommandButton3

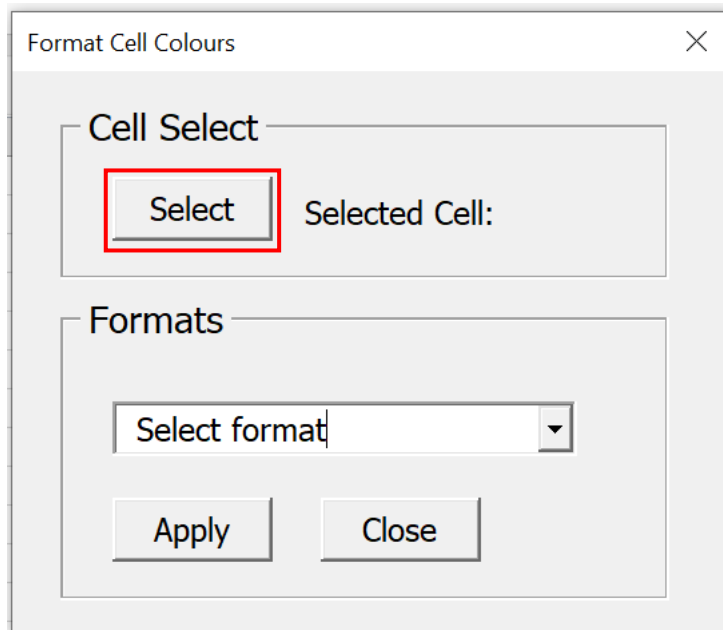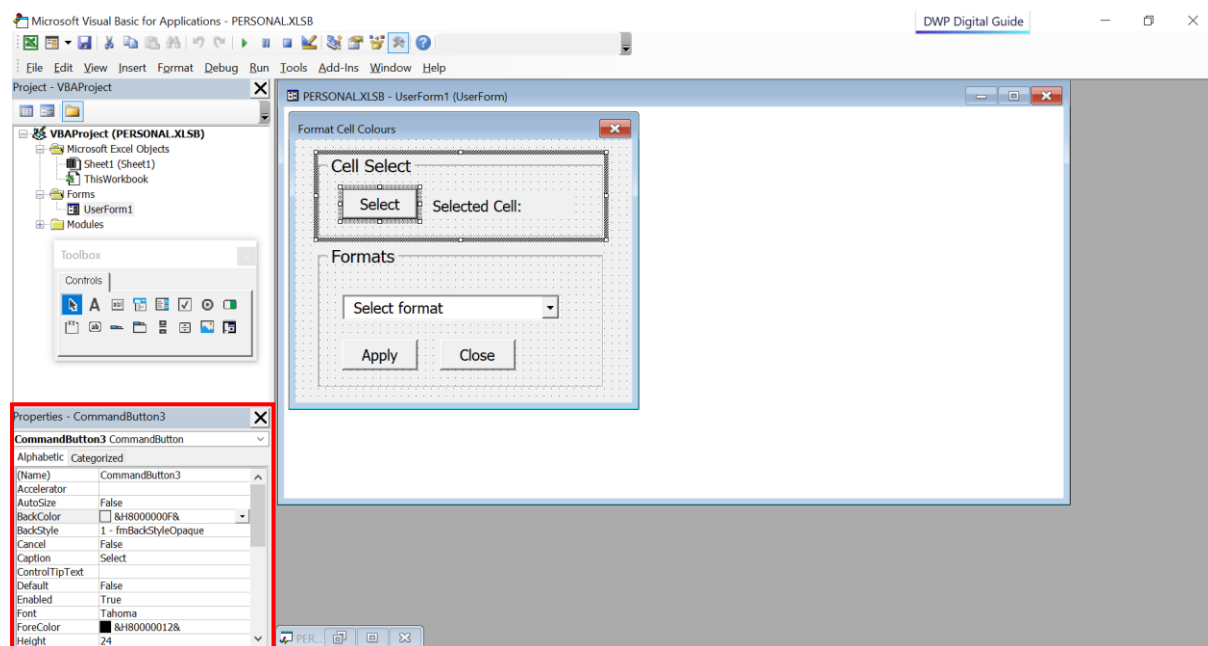|	
(Name)	CommandButton3
Accelerator	
AutoSize	False
BackColor	&H8000000F&
BackStyle	1 - fmBackStyleOpaque
Cancel	False
Caption	Select
ControlTipText	
Default	False
Enabled	True
Font	Tahoma
ForeColor	&H80000012&
Height	24
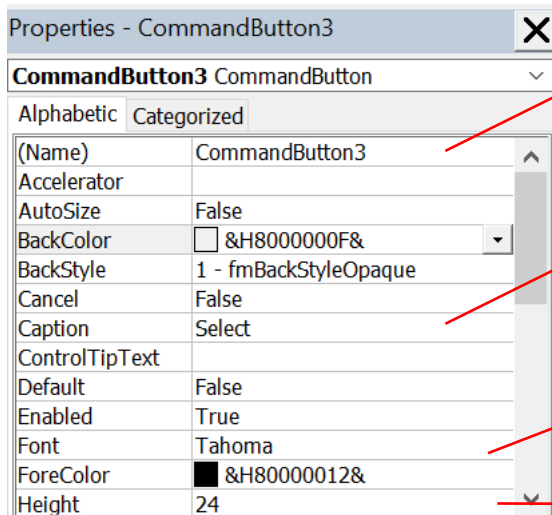
This is the "variable" name. Change this to make it easier for you to reference in code.

Either change this or directly edit the text from the button box to change the text.

Click on this to change Font and text size.

You can change the height of the button here.

Interface Reference

Here are the Names given to the UserForm objects for easy reference to the code.

Format Cell Colours

Frame2

Cell Select

CommandButton3

Select   Selected Cell:

Label2

Formats

Frame1

Label1

Select format

ComboBox1

Apply   Close

CommandButton2

CommandButton1
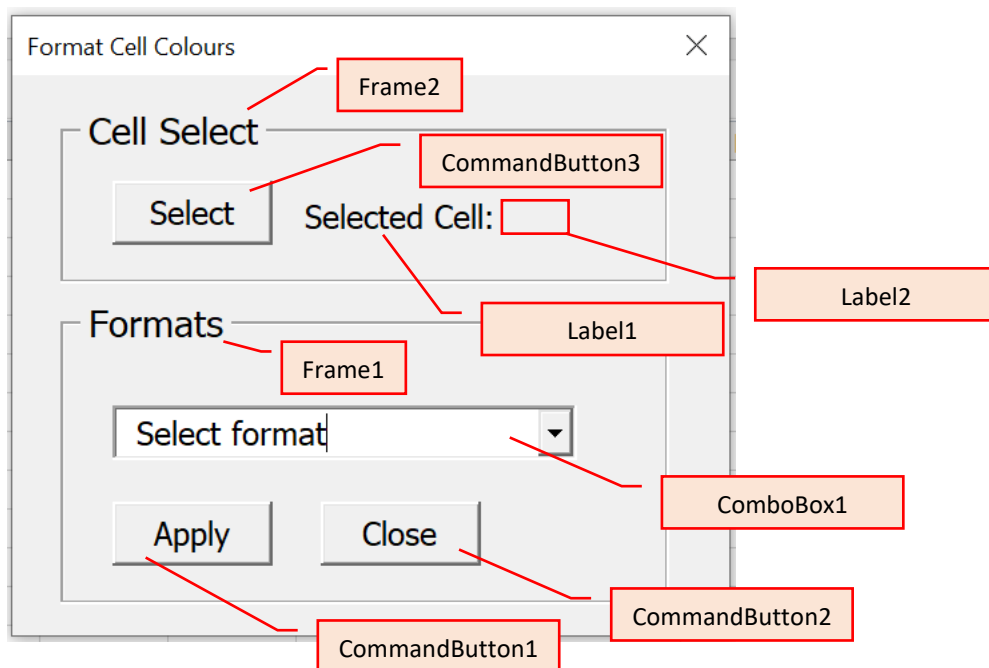
The Macro

Making the interface pop-up

Created a module that will make the UserForm pop-up by clicking on the macro button.

This in turn will allow the interface to come up.

To look at the UserForm code, right click on the UserForm and select "View Code".

In order to get to the "Private Sub UserForm_Initialize()" portion of the code, click on the dropdown to look for "initialize".

Microsoft Visual Basic for Applications - PERSONAL.XLSB - [UserForm1 (Code)]

File Edit View Insert Format Debug Run Tools Add-Ins Window Help

Ln 132, Col 1

Project - VBAProject

VBAProject (PERSONAL.XLSB)
- Microsoft Excel Objects
  - Sheet1 (Sheet1)
  - ThisWorkbook
- Forms
  - UserForm1
- Modules
  - convert_text_to_number
  - copy_formulas
  - copy_namelist
  - iterate_through_payroll
  - Module1
  - msg_box_with_cancel_option
  - userform1_popup

UserForm

(General)
ComboBox1
CommandButton1
CommandButton2
CommandButton3
Frame1
Frame2
Label1
Label2
UserForm

Initialize

```vba
' if user clicks on the "Close" button
Private Sub CommandButton2_Click()
    End
End Sub

Private Sub CommandButton3_Click()
    Dim selectCell As Range
    On Error Resume Next

    Set selectCell = Application.InputBox("Select topmost cell (not including header) in column.", Type:=8)

    On Error GoTo 0

    If selectCell Is Nothing Then
    'This will end the select without errors.
    Else
    'MsgBox selectCell.Address
    Range(selectCell.Address).Select
    'The "(False, False)" portion is to disable the "$" sign
    Label2.Caption = selectCell.Address(False, False)

    End If

End Sub

Private Sub UserForm_Initialize()
    'this is to populate the dropdown
    ComboBox1.List = Array("Open/Closed", "Pass/Fail/KIV", "Priority High", "Clear Formatting")
End Sub
```
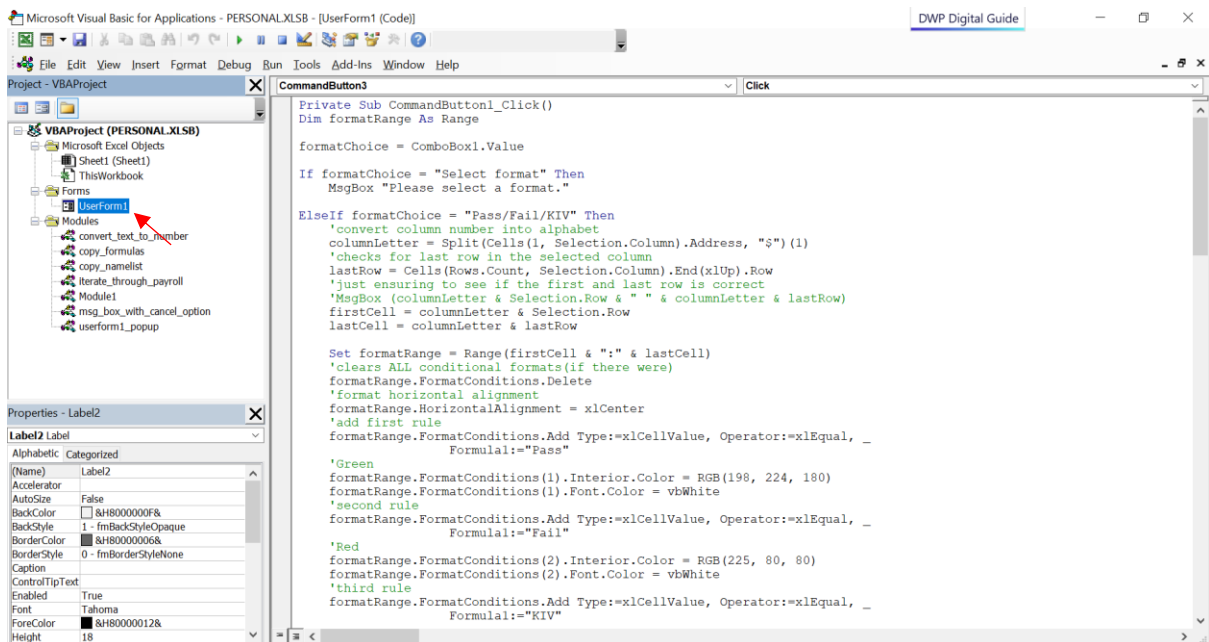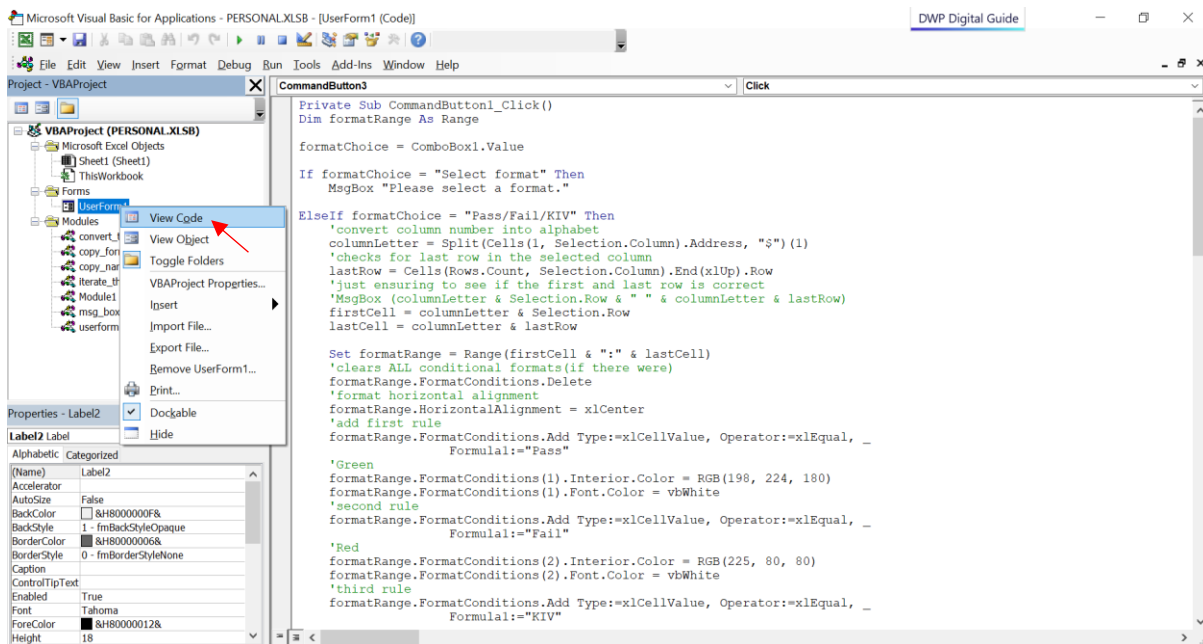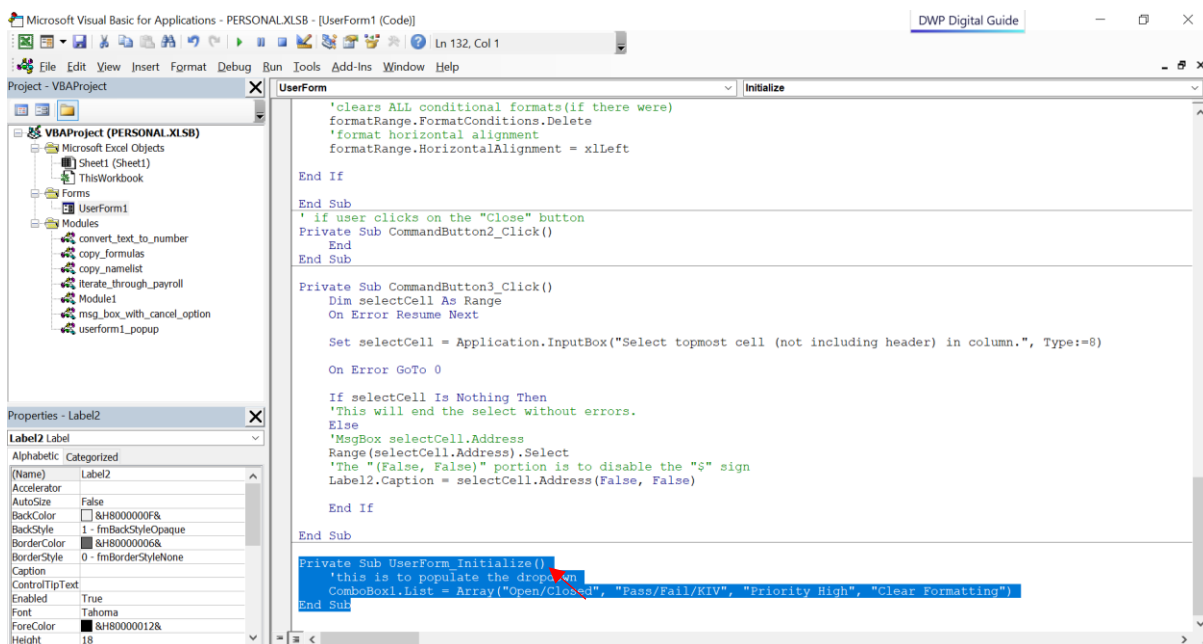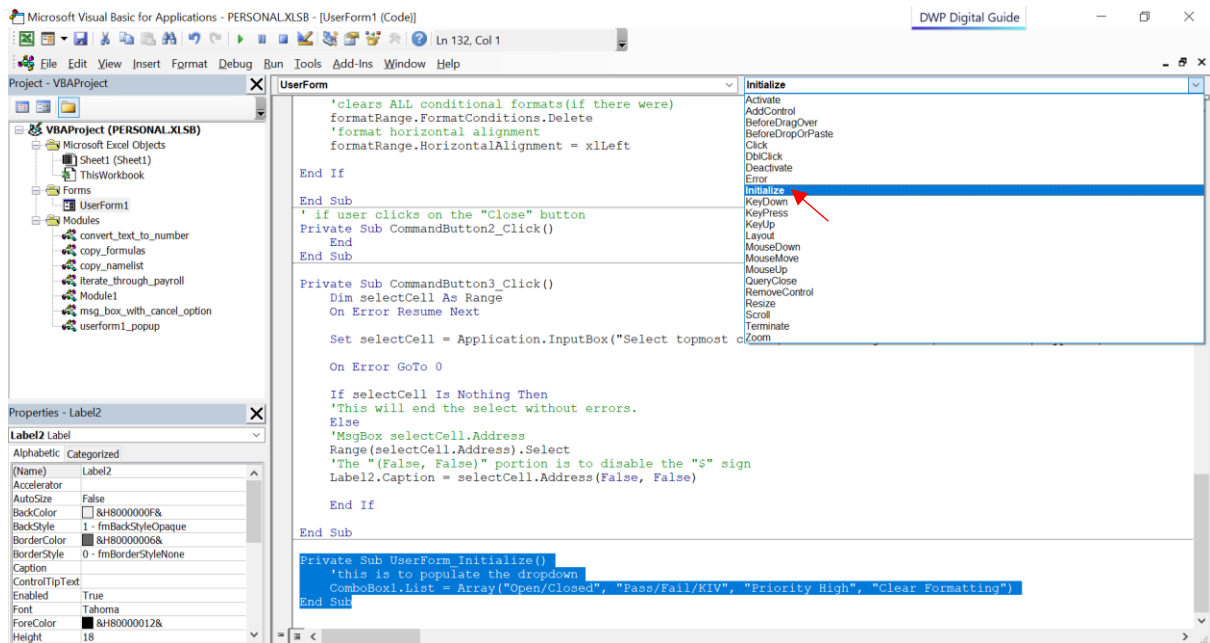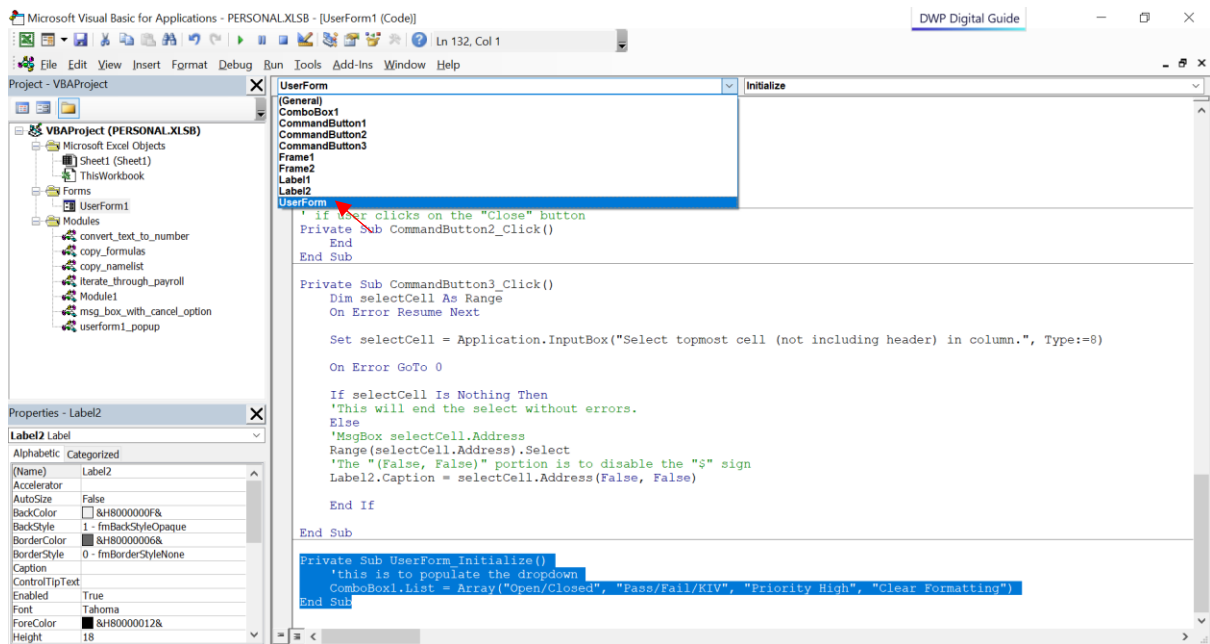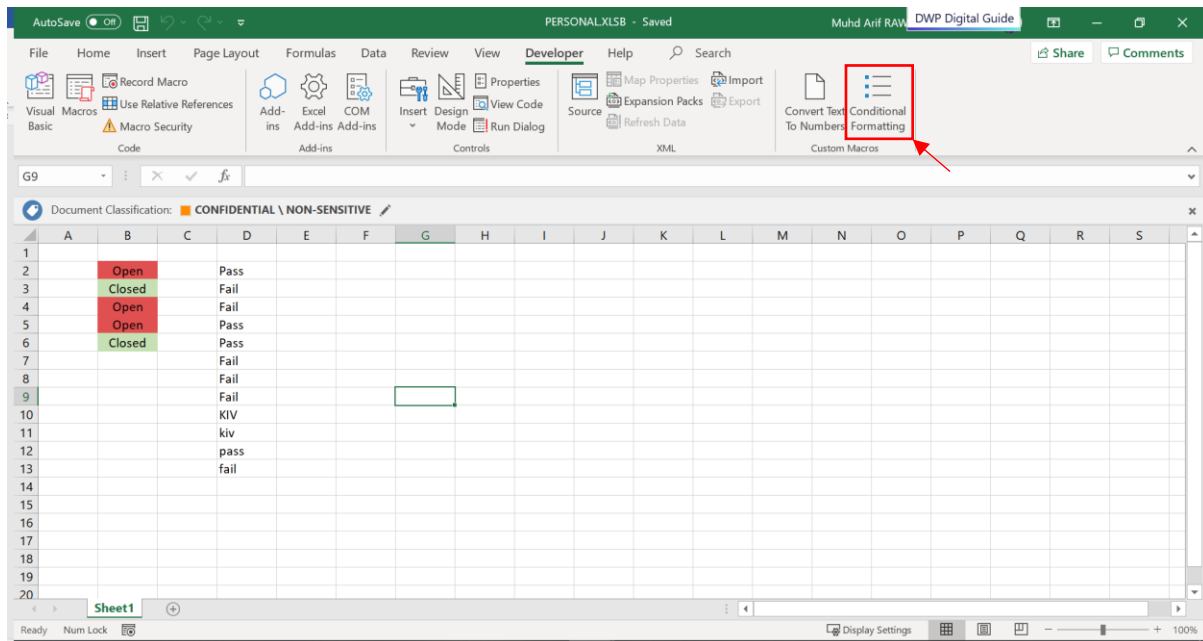
Properties - Label2

Label2 Label

Alphabetic | Categorized

(Name)	Label2
Accelerator	
AutoSize	False
BackColor	&H8000000F&
BackStyle	1 - fmBackStyleOpaque
BorderColor	&H80000006&
BorderStyle	0 - fmBorderStyleNone
Caption	
ControlTipText	
Enabled	True
Font	Tahoma
ForeColor	&H80000012&
Height	18

Microsoft Visual Basic for Applications - PERSONAL.XLSB - [UserForm1 (Code)]

File Edit View Insert Format Debug Run Tools Add-Ins Window Help

Ln 132, Col 1

Project - VBAProject

VBAProject (PERSONAL.XLSB)
- Microsoft Excel Objects
  - Sheet1 (Sheet1)
  - ThisWorkbook
- Forms
  - UserForm1
- Modules
  - convert_text_to_number
  - copy_formulas
  - copy_namelist
  - iterate_through_payroll
  - Module1
  - msg_box_with_cancel_option
  - userform1_popup

UserForm

Initialize

Activate
AddControl
BeforeDragOver
BeforeDropOrPaste
Click
DblClick
Deactivate
Error
Initialize
KeyDown
KeyPress
KeyUp
Layout
MouseDown
MouseMove
MouseUp
QueryClose
RemoveControl
Resize
Scroll
Terminate
Zoom

```vba
    'clears ALL conditional formats(if there were)
    formatRange.FormatConditions.Delete
    'format horizontal alignment
    formatRange.HorizontalAlignment = xlLeft

    End If

End Sub

' if user clicks on the "Close" button
Private Sub CommandButton2_Click()
    End
End Sub

Private Sub CommandButton3_Click()
    Dim selectCell As Range
    On Error Resume Next

    Set selectCell = Application.InputBox("Select topmost c

    On Error GoTo 0

    If selectCell Is Nothing Then
    'This will end the select without errors.
    Else
    'MsgBox selectCell.Address
    Range(selectCell.Address).Select
    'The "(False, False)" portion is to disable the "$" sign
    Label2.Caption = selectCell.Address(False, False)

    End If

End Sub

Private Sub UserForm_Initialize()
    'this is to populate the dropdown
    ComboBox1.List = Array("Open/Closed", "Pass/Fail/KIV", "Priority High", "Clear Formatting")
End Sub
```

Properties - Label2

Label2 Label

Alphabetic | Categorized

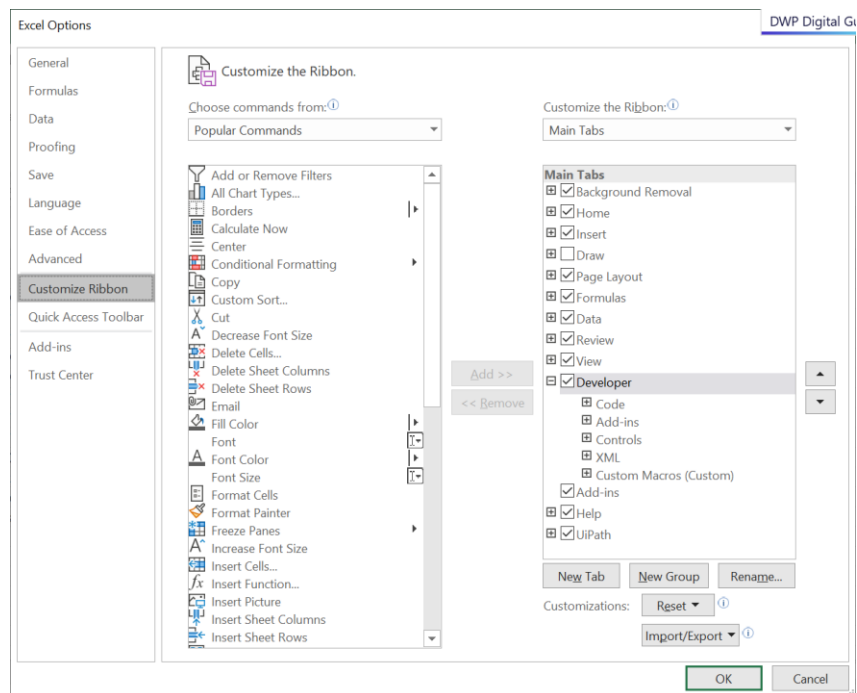|	
(Name)	Label2
Accelerator	
AutoSize	False
BackColor	&H8000000F&
BackStyle	1 - fmBackStyleOpaque
BorderColor	&H80000006&
BorderStyle	0 - fmBorderStyleNone
Caption	
ControlTipText	
Enabled	True
Font	Tahoma
ForeColor	&H80000012&
Height	18

Placing the Macro for usage

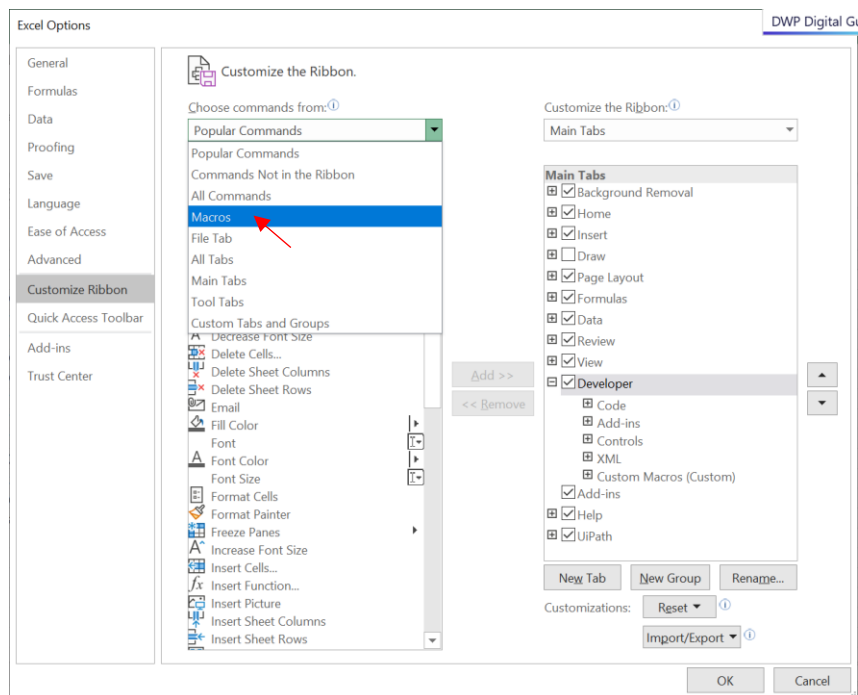In order for the Macro to be used often, you can create a custom ribbon to be placed in the Tab.

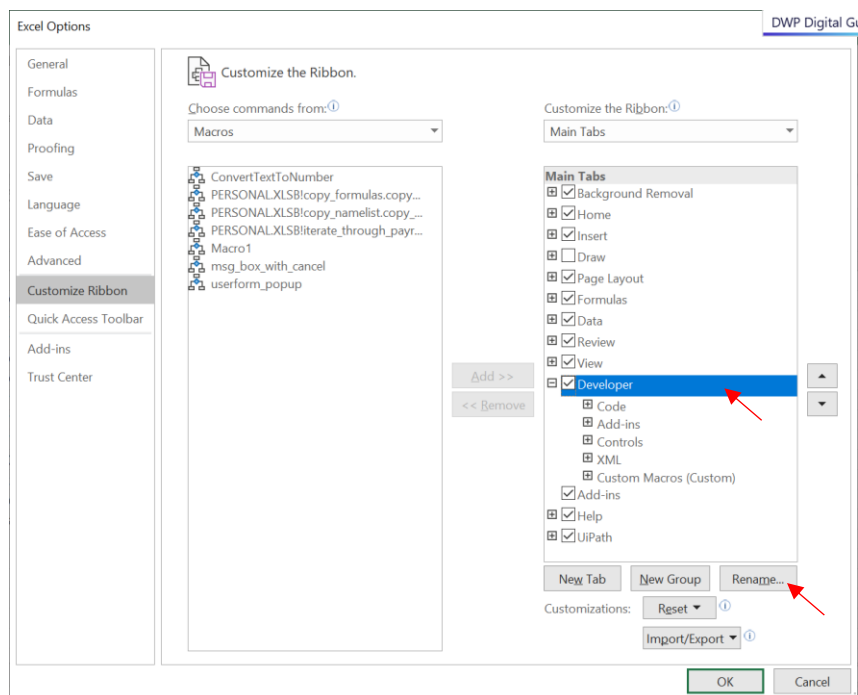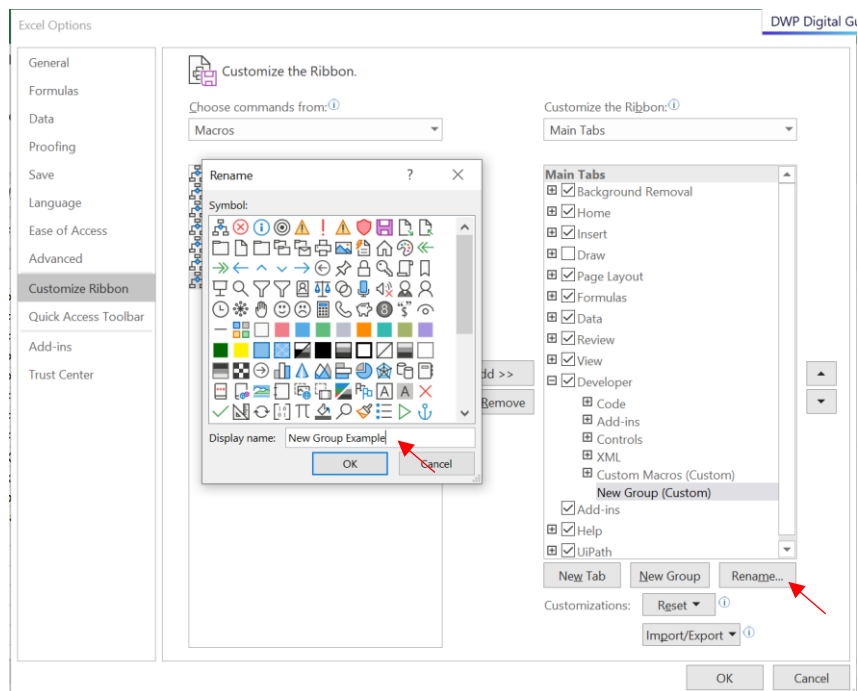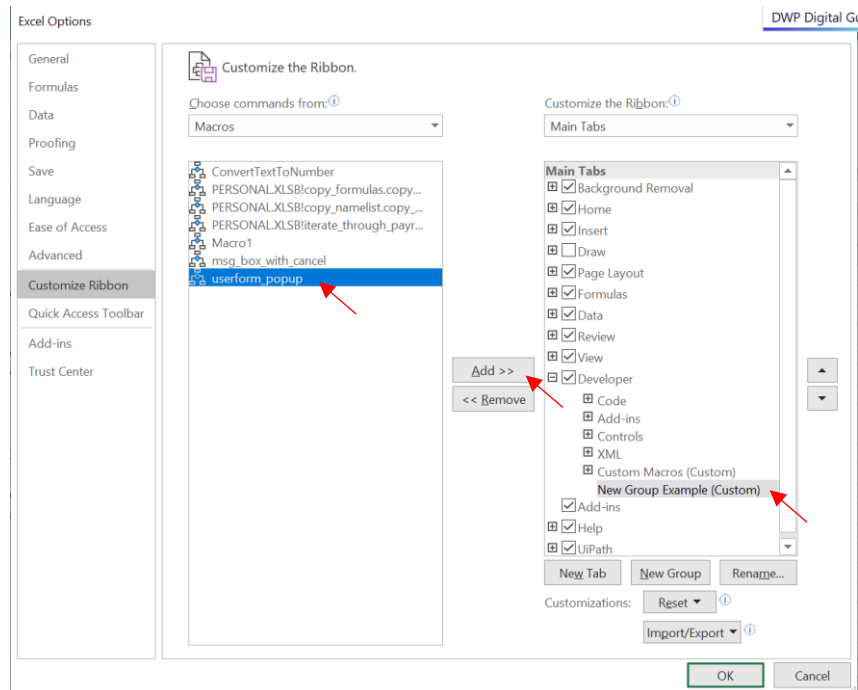Go to File > Options > Customize Ribbon.

Click on Macros

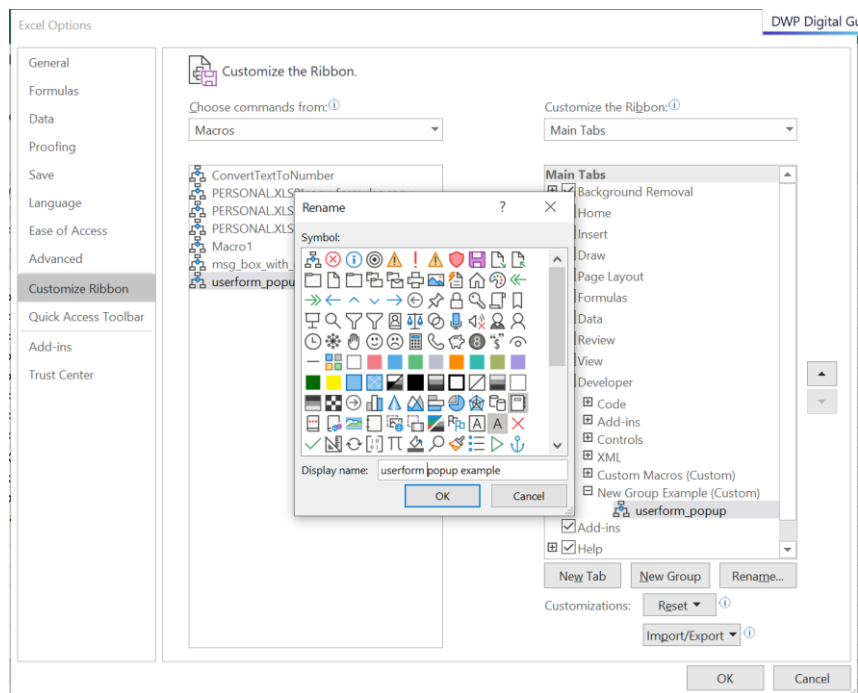Click on "Developer" then "New Group".

You can rename and even add an icon for it.

Select the Macro you wish to put in the new group. Ensure that you select both the macro and group you wish to move it into. The click on "Add >>".

You can add icon and rename the Macro for easy reference.

Click on "Okay" when you're done. The new Macro should be in the new tab now.