

VBA - USER FORMS

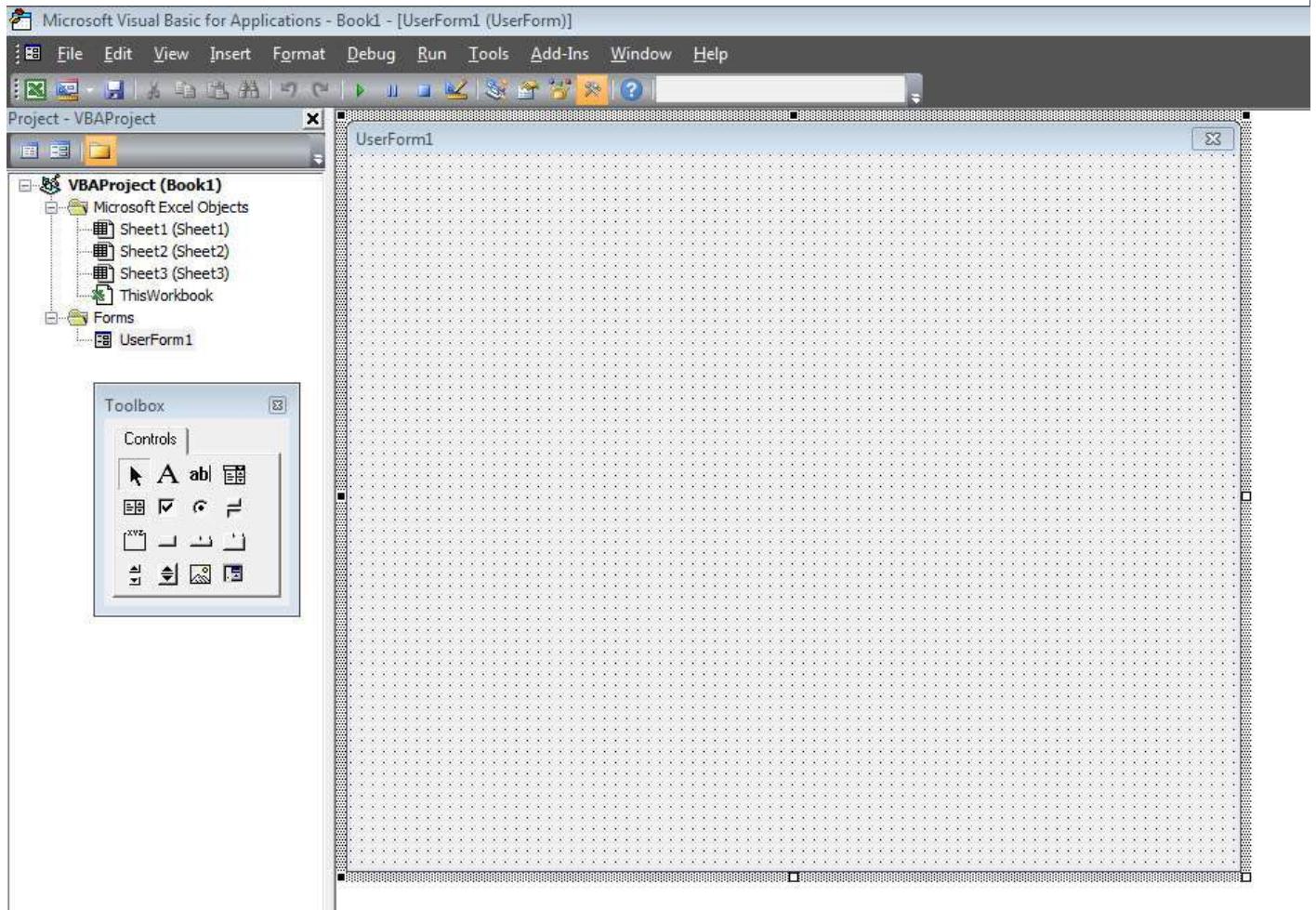
http://www.tutorialspoint.com/vba/vba_userforms.htm

Copyright © tutorialspoint.com

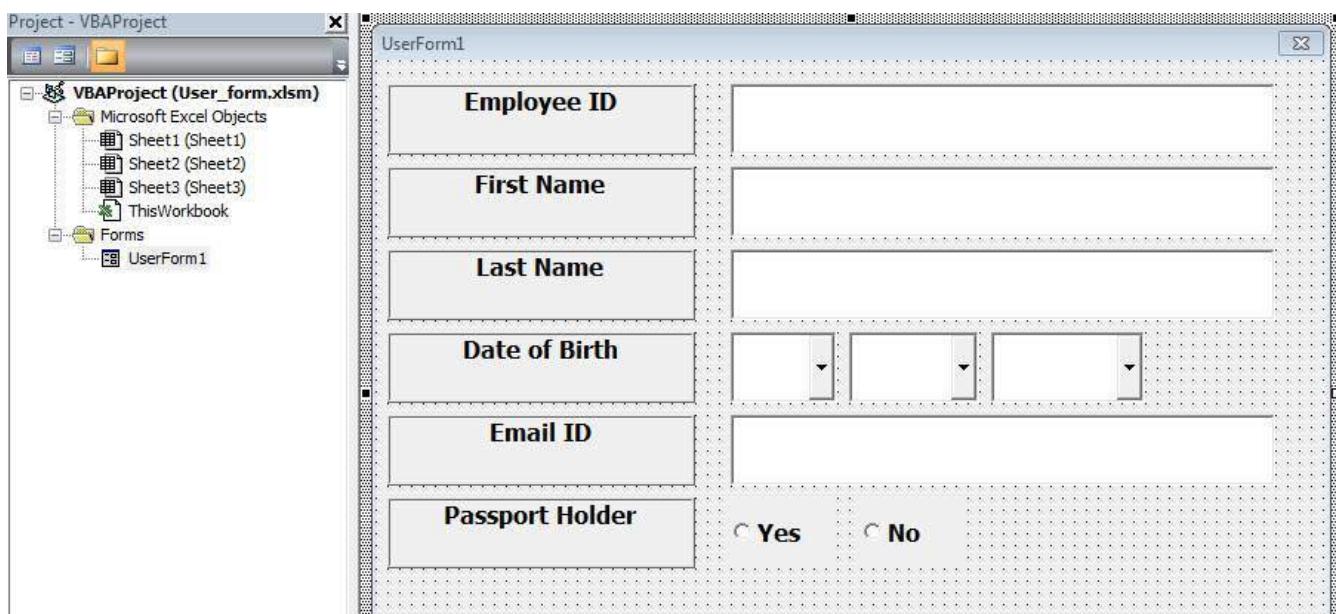
VBA - User Forms

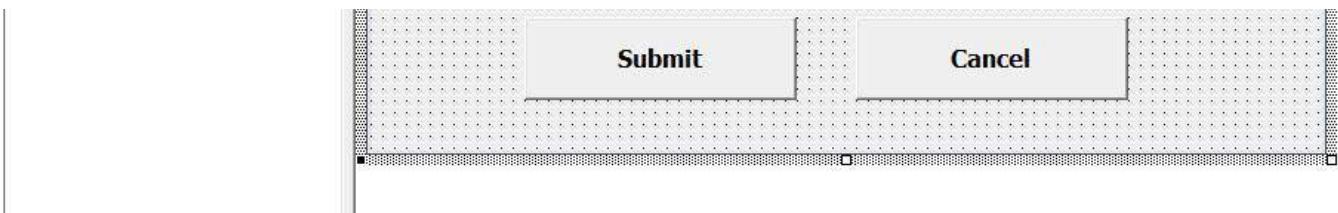
A UserForm is a custom-built dialog box that makes a user data entry more controllable for you and easier for the user. In this chapter, we will design a Simple form and add data into excel.

Step 1 : Navigate to VBA Window by pressing Alt+F11 and Navigate to "Insert" Menu and select "User Form". Upon selecting, user form is displayed below.



Step 2 : Now let us design the forms using the given controls.





Step 3 : After adding each controls, the controls has to be named. Caption corresponds to what appears on the form and name corresponds to the logical name which will be appearing while we write VBA code for that element.

The screenshot shows the Microsoft Excel VBA environment. On the left is the Project Explorer window showing 'VBAProject (User_form.xlsm)' with 'Employee ID' selected. To its right is the Properties window for a control named 'empid'. A yellow callout bubble with the text 'Logical Name' points to the 'Caption' field, which is set to 'Employee ID'. A red arrow also points from the caption text in the callout to the 'Caption' field in the properties list. The main area shows a UserForm titled 'Employee ID' with a single label control.

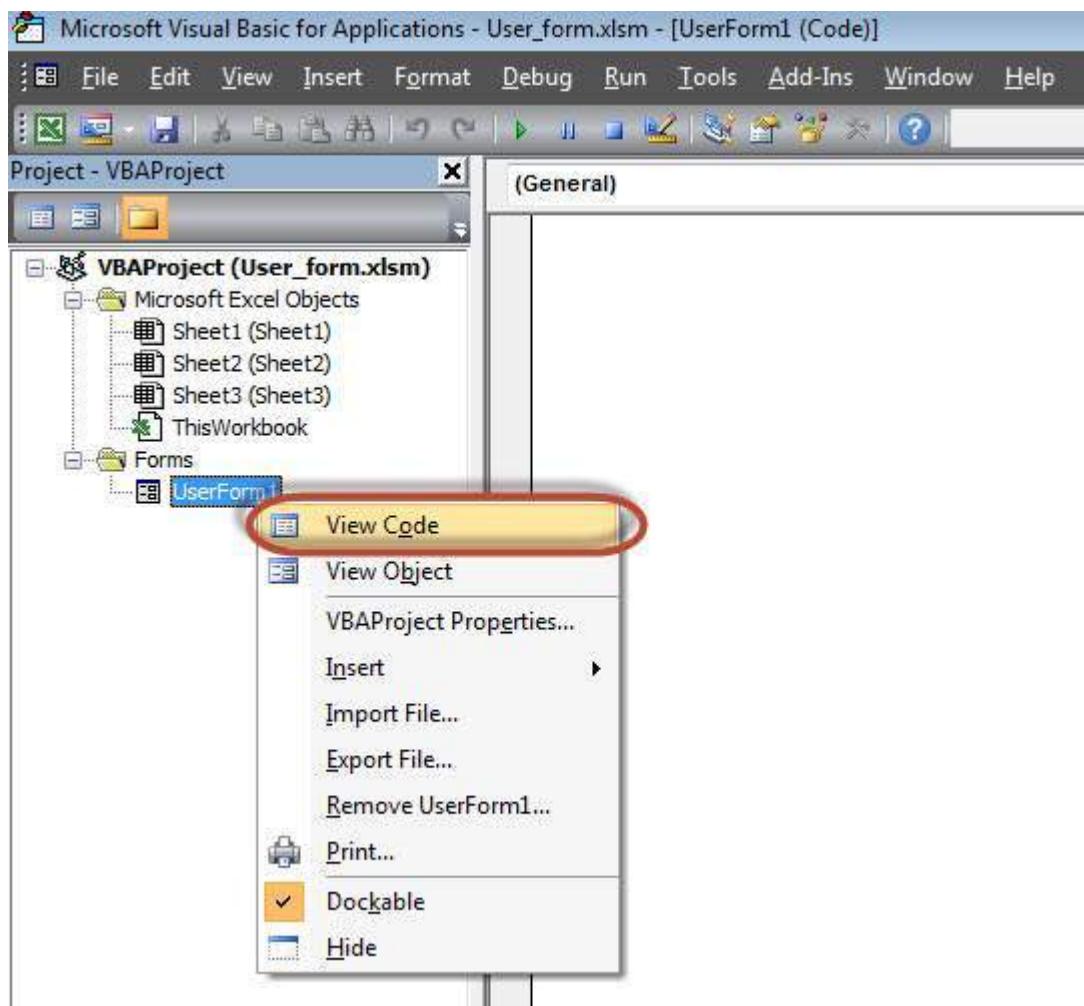
Control	Logical Name	Caption
From	frmempform	Employee Form
Employee ID Label Box	empid	Employee ID
firstname Label Box	firstname	First Name
lastname Label Box	lastname	Last Name
dob Label Box	dob	Date of Birth
mailid Label Box	mailid	Email ID
Passportholder Label Box	Passportholder	Passport Holder
Emp ID Text Box	txtempid	NOT Applicable
First Name Text Box	txtfirstname	NOT Applicable
Last Name Text Box	txtlastname	NOT Applicable
Email ID Text Box	txtemailid	NOT Applicable

Step 4 : Below are names against each one of the added controls.

Control	Logical Name	Caption
From	frmempform	Employee Form
Employee ID Label Box	empid	Employee ID
firstname Label Box	firstname	First Name
lastname Label Box	lastname	Last Name
dob Label Box	dob	Date of Birth
mailid Label Box	mailid	Email ID
Passportholder Label Box	Passportholder	Passport Holder
Emp ID Text Box	txtempid	NOT Applicable
First Name Text Box	txtfirstname	NOT Applicable
Last Name Text Box	txtlastname	NOT Applicable
Email ID Text Box	txtemailid	NOT Applicable

Date Combo Box	cmbdate	NOT Applicable
Month Combo Box	cmbmonth	NOT Applicable
Year Combo Box	cmbyear	NOT Applicable
Yes Radio Button	radioyes	Yes
No Radio Button	radiono	No
Submit Button	btnsubmit	Submit
Cancel Button	btncancel	Cancel

Step 5 : Now we will add code for the form load event by performing right click on the form and selecting 'View Code'.



Step 6 : Select userform from the objects drop down and select 'Initialize' method as shown below.

The screenshot shows the Microsoft Visual Basic for Applications code editor. The dropdown menu at the top has 'UserForm' selected (circled with a red circle). In the code window, the following VBA code is displayed:

```

Private Sub UserForm_Initialize()

End Sub

```

The 'Initialize' method is circled with a red circle in the bottom right corner of the code editor.

Step 7 : Upon Loading the Form we should ensure that the text boxes are cleared, Drop down boxes are filled and Radio buttons are resetted

```
Private Sub UserForm_Initialize()

'Empty Emp ID Text box and Set the Cursor
txtempid.Value = ""
txtempid.SetFocus

'Empty all other text box fields
txtfirstname.Value = ""
txtlastname.Value = ""
txtemailid.Value = ""

'Clear All Date of Birth Related Fields
cmbdate.Clear
cmbmonth.Clear
cmbyear.Clear

'Fill Date Drop Down box - Takes 1 to 31
With cmbdate
    .AddItem "1"
    .AddItem "2"
    .AddItem "3"
    .AddItem "4"
    .AddItem "5"
    .AddItem "6"
    .AddItem "7"
    .AddItem "8"
    .AddItem "9"
    .AddItem "10"
    .AddItem "11"
    .AddItem "12"
    .AddItem "13"
    .AddItem "14"
    .AddItem "15"
    .AddItem "16"
    .AddItem "17"
    .AddItem "18"
    .AddItem "19"
    .AddItem "20"
    .AddItem "21"
    .AddItem "22"
    .AddItem "23"
    .AddItem "24"
    .AddItem "25"
    .AddItem "26"
    .AddItem "27"
    .AddItem "28"
    .AddItem "29"
    .AddItem "30"
    .AddItem "31"
End With

'Fill Month Drop Down box - Takes Jan to Dec
With cmbmonth
    .AddItem "JAN"
    .AddItem "FEB"
    .AddItem "MAR"
    .AddItem "APR"
    .AddItem "MAY"
    .AddItem "JUN"
    .AddItem "JUL"
    .AddItem "AUG"
    .AddItem "SEP"
    .AddItem "OCT"
    .AddItem "NOV"
    .AddItem "DEC"
End With
```

```

'Fill Year Drop Down box - Takes 1980 to 2014
With cmbyear
    .AddItem "1980"
    .AddItem "1981"
    .AddItem "1982"
    .AddItem "1983"
    .AddItem "1984"
    .AddItem "1985"
    .AddItem "1986"
    .AddItem "1987"
    .AddItem "1988"
    .AddItem "1989"
    .AddItem "1990"
    .AddItem "1991"
    .AddItem "1992"
    .AddItem "1993"
    .AddItem "1994"
    .AddItem "1995"
    .AddItem "1996"
    .AddItem "1997"
    .AddItem "1998"
    .AddItem "1999"
    .AddItem "2000"
    .AddItem "2001"
    .AddItem "2002"
    .AddItem "2003"
    .AddItem "2004"
    .AddItem "2005"
    .AddItem "2006"
    .AddItem "2007"
    .AddItem "2008"
    .AddItem "2009"
    .AddItem "2010"
    .AddItem "2011"
    .AddItem "2012"
    .AddItem "2013"
    .AddItem "2014"
End With

'Reset Radio Button. Set it to False when form loads.
radioyes.Value = False
radiono.Value = False

End Sub

```

Step 8 : Now we need to add code to the Submit button. Upon Clicking on submit button user Should be able to Add the values into the worksheet.

```

Private Sub btnsubmit_Click()
Dim emptyRow As Long

'Make Sheet1 active
Sheet1.Activate

'Determine emptyRow
emptyRow = WorksheetFunction.CountA(Range("A:A")) + 1

'Transfer information
Cells(emptyRow, 1).Value = txtempid.Value
Cells(emptyRow, 2).Value = txtfirstname.Value
Cells(emptyRow, 3).Value = txtlastname.Value
Cells(emptyRow, 4).Value = cmbdate.Value & "/" & cmbmonth.Value & "/" & cmbyear.Value
Cells(emptyRow, 5).Value = txtemailid.Value

If radioyes.Value = True Then
    Cells(emptyRow, 6).Value = "Yes"
Else
    Cells(emptyRow, 6).Value = "No"
End If

```

```
End Sub
```

Step 9 : Now add a method to close the form when user clicks on cancel button.

```
Private Sub btncancel_Click()  
    Unload Me  
End Sub
```

Step 10 : Now Let us Execute the Form by clicking on "run" Button. Enter values into the form and click 'Submit' button. Automatically the values would flow into the worksheet as shown below.

	A	B	C	D	E	F
1	Employee ID	First Name	Last Name	Date of Birth	Email ID	Passport Holder
2	100 Jim		Smith		1-Feb-82 jim@Tutorialspoint.com	Yes
3	101 Jack		Rose		19-Aug-88 Jack.Rose@Tutorialspoint.com	No
4	102 Cathy		Nelson		27-Jul-86 cathy@tutorialspoint.com	No
5						
6						
7						
8						
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						
19						
20						
21						
22						
23						
24						
25						
26						
27						
28						
29						
30						
31						
32						

Employee Form

The screenshot shows a Microsoft Excel spreadsheet with a data table and a modal dialog box. The dialog box is titled 'Employee Form' and contains fields for Employee ID, First Name, Last Name, Date of Birth (with dropdown menus for day, month, and year), Email ID, and Passport Holder (with radio buttons for Yes and No, where No is selected). A red arrow points from the bottom right of the dialog box towards the top right of the spreadsheet table, indicating the flow of data from the form to the sheet.