**Module 11: Forms & Controls**

* **Introduction to UserForms**

**UserForms** are custom dialog boxes in Excel VBA that allow users to interact with the program through graphical interfaces such as textboxes, buttons, combo boxes, etc.

They’re used for:

* Data entry
* User input and validation
* Running automated tasks with user choices

**To create a UserForm:**

1. Press Alt + F11 → open VBA Editor
2. Go to **Insert → UserForm**
3. A blank form will appear — you can add controls using the **Toolbox**

* **Adding & Configuring Controls**

Common Controls:

| **Control** | **Description** |
| --- | --- |
| **Label** | Displays static text or headings |
| **TextBox** | Allows user text input |
| **ComboBox** | Drop-down list of predefined options |
| **CommandButton** | Executes code when clicked |
| **CheckBox / OptionButton** | Allows Yes/No or multiple choice input |
| **ListBox** | Displays a list of selectable items |

**Example Layout:**

| **Control** | **Name** | **Purpose** |
| --- | --- | --- |
| Label | lblName | Caption: "Enter Your Name:" |
| TextBox | txtName | User enters their name |
| ComboBox | cboDepartment | Dropdown with departments |
| CommandButton | btnSubmit | Button caption: “Submit” |

* **Event-Driven Programming with Controls**

Event-driven VBA means your VBA code runs automatically in response to user actions or system events, rather than you manually running macros.

Each control can **trigger events**, like:

* Click (for buttons)
* Change (for textboxes)
* Initialize (for the form itself)

**Example: Simple Data Entry Form**

**Steps to Try**

1. In Excel, open VBA Editor (Alt + F11)
2. Insert a **UserForm**
3. Add:
   * Label (lblName)
   * TextBox (txtName)
   * ComboBox (cboDept)
   * CommandButton (btnSubmit)
4. Double-click the form and paste the code below 👇

Code Example

' ==== UserForm Code ====

Private Sub UserForm\_Initialize()

' Populate ComboBox with departments when form loads

cboDept.AddItem "Finance"

cboDept.AddItem "HR"

cboDept.AddItem "IT"

cboDept.AddItem "Marketing"

End Sub

Private Sub btnSubmit\_Click()

Dim name As String

Dim dept As String

name = txtName.Text

dept = cboDept.Text

If name = "" Or dept = "" Then

MsgBox "Please enter your name and select a department.", vbExclamation

Exit Sub

End If

' Write data to Excel sheet

Dim ws As Worksheet

Set ws = ThisWorkbook.Sheets("Sheet1")

Dim nextRow As Long

nextRow = ws.Cells(ws.Rows.Count, "A").End(xlUp).Row + 1

ws.Cells(nextRow, 1).Value = name

ws.Cells(nextRow, 2).Value = dept

MsgBox "Data submitted successfully!", vbInformation

Unload Me

End Sub

**Add a Module to Open the Form**

1. Go to **Insert → Module**
2. Paste this code:

Sub OpenEmployeeForm()

UserForm1.Show

End Sub

1. Go back to Excel and press **Alt + F8**, select **OpenEmployeeForm**, and click **Run**  
   👉 Your form will appear!

**Real Life Use OF UserForm Project (Employee Registration System)**

This is a **step-by-step practical project** that lets you:  
✅ Add new employees  
✅ Search employees  
✅ Update existing details  
✅ Delete records

You’ll learn professional-level **event-driven VBA with forms & Excel integration** — exactly how HR tools or small CRMs work internally.

**🧩 STEP 1: Create the Excel Setup**

Open a new Excel workbook and rename **Sheet1** to:  
👉 EmployeeData

Create these **headers** in Row 1:

| **A** | **B** | **C** | **D** |
| --- | --- | --- | --- |
| Emp ID | Name | Department | Salary |

**🧩 STEP 2: Create a UserForm**

1. Press **Alt + F11** → VBA editor
2. Go to **Insert → UserForm**
3. Rename it (in Properties window) to frmEmployee

**🎛️ STEP 3: Add Controls on the Form**

Use the **Toolbox** to place these controls neatly:

|  |  |  |
| --- | --- | --- |
| **Control** | **Name** | **Caption / Purpose** |
| Label | lblEmpID | “Employee ID:” |
| TextBox | txtEmpID | Enter ID |
| Label | lblName | “Name:” |
| TextBox | txtName | Enter name |
| Label | lblDept | “Department:” |
| ComboBox | cboDept | Choose department |
| Label | lblSalary | “Salary:” |
| TextBox | txtSalary | Enter salary |
| CommandButton | btnAdd | “Add” |
| CommandButton | btnSearch | “Search” |
| CommandButton | btnUpdate | “Update” |
| CommandButton | btnDelete | “Delete” |
| CommandButton | btnClear | “Clear” |

**🧠 STEP 4: Add This Code to frmEmployee**

Double-click your form → paste this full code 👇

Private Sub UserForm\_Initialize()

' Populate department options

cboDept.AddItem "Finance"

cboDept.AddItem "HR"

cboDept.AddItem "IT"

cboDept.AddItem "Marketing"

End Sub

Private Sub btnAdd\_Click()

Dim ws As Worksheet

Set ws = ThisWorkbook.Sheets("EmployeeData")

If txtEmpID.Value = "" Or txtName.Value = "" Or cboDept.Value = "" Or txtSalary.Value = "" Then

MsgBox "Please fill all fields before adding.", vbExclamation

Exit Sub

End If

Dim nextRow As Long

nextRow = ws.Cells(ws.Rows.Count, "A").End(xlUp).Row + 1

ws.Cells(nextRow, 1).Value = txtEmpID.Value

ws.Cells(nextRow, 2).Value = txtName.Value

ws.Cells(nextRow, 3).Value = cboDept.Value

ws.Cells(nextRow, 4).Value = txtSalary.Value

MsgBox "Employee added successfully!", vbInformation

Call btnClear\_Click

End Sub

Private Sub btnSearch\_Click()

Dim ws As Worksheet

Set ws = ThisWorkbook.Sheets("EmployeeData")

Dim empID As String

empID = txtEmpID.Value

Dim r As Range

If empID = "" Then

MsgBox "Please enter Employee ID to search.", vbExclamation

Exit Sub

End If

Set r = ws.Range("A:A").Find(empID, LookIn:=xlValues, LookAt:=xlWhole)

If Not r Is Nothing Then

txtName.Value = ws.Cells(r.Row, 2).Value

cboDept.Value = ws.Cells(r.Row, 3).Value

txtSalary.Value = ws.Cells(r.Row, 4).Value

MsgBox "Record found!", vbInformation

Else

MsgBox "Employee ID not found.", vbCritical

End If

End Sub

Private Sub btnUpdate\_Click()

Dim ws As Worksheet

Set ws = ThisWorkbook.Sheets("EmployeeData")

Dim r As Range

Set r = ws.Range("A:A").Find(txtEmpID.Value, LookIn:=xlValues, LookAt:=xlWhole)

If Not r Is Nothing Then

ws.Cells(r.Row, 2).Value = txtName.Value

ws.Cells(r.Row, 3).Value = cboDept.Value

ws.Cells(r.Row, 4).Value = txtSalary.Value

MsgBox "Record updated successfully!", vbInformation

Call btnClear\_Click

Else

MsgBox "Employee ID not found.", vbCritical

End If

End Sub

Private Sub btnDelete\_Click()

Dim ws As Worksheet

Set ws = ThisWorkbook.Sheets("EmployeeData")

Dim r As Range

Set r = ws.Range("A:A").Find(txtEmpID.Value, LookIn:=xlValues, LookAt:=xlWhole)

If Not r Is Nothing Then

ws.Rows(r.Row).Delete

MsgBox "Record deleted successfully!", vbInformation

Call btnClear\_Click

Else

MsgBox "Employee ID not found.", vbCritical

End If

End Sub

Private Sub btnClear\_Click()

txtEmpID.Value = ""

txtName.Value = ""

cboDept.Value = ""

txtSalary.Value = ""

End Sub

**🧩 STEP 5: Add a Module to Show the Form**

Insert → **Module** → paste this:

Sub OpenEmployeeForm()

frmEmployee.Show

End Sub

**🧩 STEP 6: Run It 🚀**

1. Go back to Excel → press **Alt + F8**
2. Choose OpenEmployeeForm → click **Run**
3. You’ll see a complete Employee Form!