```
Private Sub Frame1_Click()
End Sub
Private Sub Label1_Click()
End Sub
Private Sub Label10_Click()
End Sub
Private Sub Label11_Click()
End Sub
Private Sub Label12_Click()
End Sub
Private Sub Label13_Click()
End Sub
Private Sub Label14_Click()
End Sub
Private Sub Label15_Click()
End Sub
Private Sub Label16_Click()
End Sub
Private Sub Label17_Click()
End Sub
Private Sub Label18_Click()
End Sub
Private Sub Label19_Click()
End Sub
Private Sub Label2_Click()
End Sub
Private Sub Label3_Click()
End Sub
Private Sub Label4_Click()
End Sub
Private Sub Label5_Click()
End Sub
Private Sub Label6_Click()
End Sub
Private Sub Label7_Click()
End Sub
Private Sub Label8_Click()
```

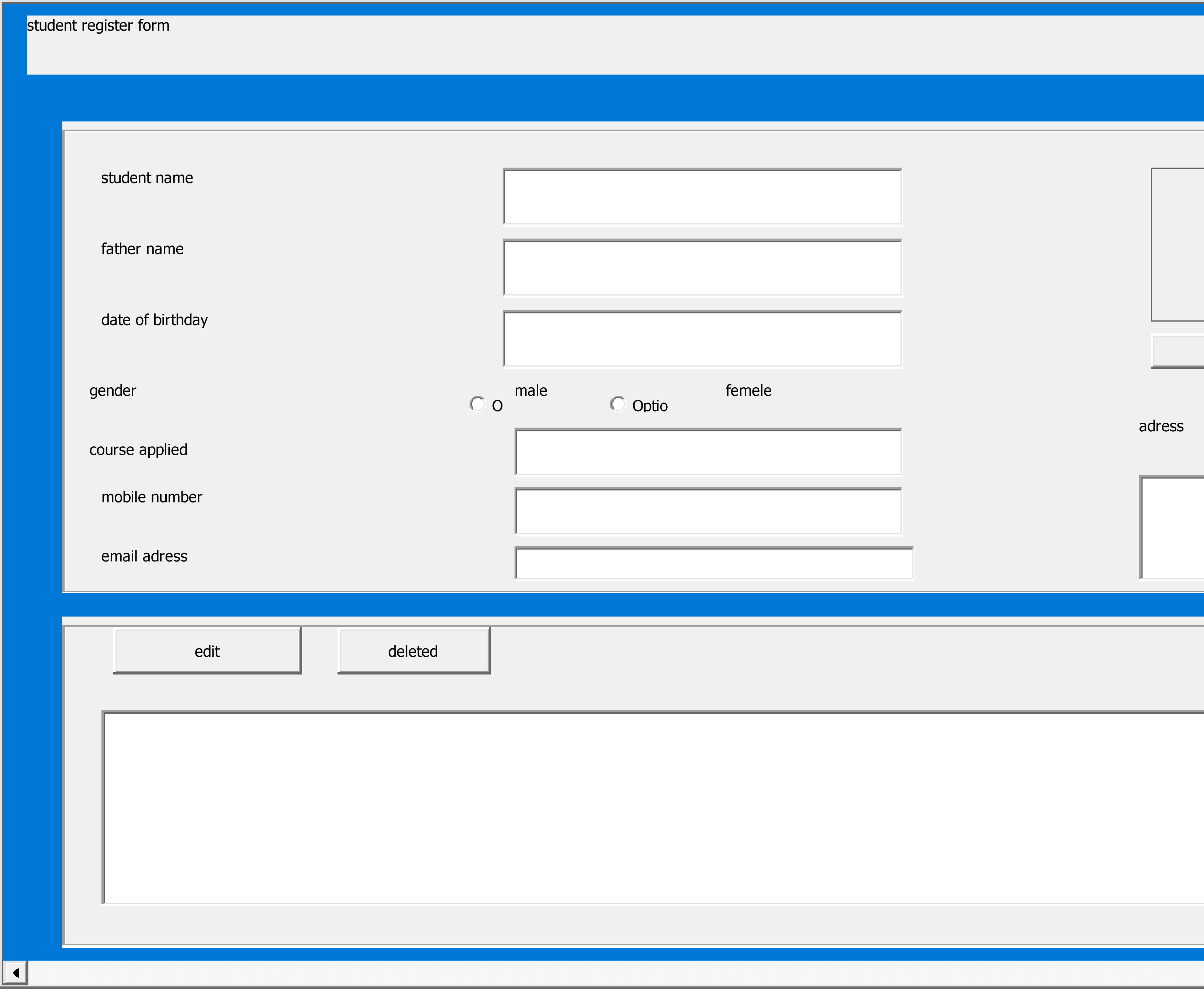
```
End Sub

Private Sub Label9_Click()

End Sub

Private Sub ListBox1_Click()

End Sub

Private Sub OptionButton1_Click()

End Sub

Private Sub OptionButton2_Click()

End Sub

Private Sub TextBox1_Change()

End Sub

Private Sub TextBox3_Change()

End Sub

Private Sub TextBox5_Change()

End Sub

Private Sub TextBox7_Change()

End Sub

Private Sub UserForm_Activate()

End Sub

Private Sub UserForm_AddControl(ByVal Control As MSForms.Control)

End Sub

Private Sub UserForm_BeforeDragOver(ByVal Cancel As MSForms.ReturnBoolean, ByVal Control As MSForms.Co
ntrol, ByVal Data As MSForms.DataObject, ByVal X As Single, ByVal Y As Single, ByVal State As MSForms.
fmDragState, ByVal Effect As MSForms.ReturnEffect, ByVal Shift As Integer)

End Sub

Private Sub UserForm_Click()

End Sub

Private Sub UserForm_Deactivate()

End Sub

Private Sub UserForm_Initialize()

End Sub

Private Sub UserForm_Layout()

End Sub

Private Sub UserForm_MouseMove(ByVal Button As Integer, ByVal Shift As Integer, ByVal X As Single, ByV
al Y As Single)

End Sub

Private Sub UserForm_MouseUp(ByVal Button As Integer, ByVal Shift As Integer, ByVal X As Single, ByVal
 Y As Single)
```

```
End Sub

Private Sub UserForm_RemoveControl(ByVal Control As MSForms.Control)

End Sub

Private Sub UserForm_Resize()

End Sub

Private Sub UserForm_Scroll(ByVal ActionX As MSForms.fmScrollAction, ByVal ActionY As MSForms.fmScroll
Action, ByVal RequestDx As Single, ByVal RequestDy As Single, ByVal ActualDx As MSForms.ReturnSingle,
ByVal ActualDy As MSForms.ReturnSingle)

End Sub

Private Sub UserForm_Terminate()

End Sub

Private Sub UserForm_Zoom(Percent As Integer)


Application.ScreenUpdating = False

Dim sDate As String

On Error Resume Next

sDate = MyCalendar.DatePicker(Me.txtDOB)

Me.txtDOB.Value = Format(sDate, "dd-mmm-yyyy")

On Error GoTo 0

Application.ScreenUpdating = True
End Sub


Private Sub imgCalendar_Click()

Application.ScreenUpdating = False

Dim sDate As String

On Error Resume Next

sDate = MyCalendar.DatePicker(Me.txtDOB)

Me.txtDOB.Value = Format(sDate, "dd-mmm-yyyy")
Sub Reset_Form()
Dim iRow As Long

With frmDataEntry

    .txtStudentName.Text = ""
    .txtStudentName.BackColor = vbWhite

    .txtFatherName.Text = ""
    .txtFatherName.BackColor = vbWhite

    .txtDOB.Text = ""
    .txtDOB.BackColor = vbWhite

    .optFemale.Value = False
    .optMale.Value = False

    .txtMobile.Value = ""
    .txtMobile.BackColor = vbWhite

    .txtEmail.Value = ""
    .txtEmail.BackColor = vbWhite
    .txtAddress.Value = ""
    .txtAddress.BackColor = vbWhite

    .txtRowNumber.Value = ""
    .txtImagePath.Value = ""

    .imgStudent.Picture = LoadPicture(vbNullString)

    .cmdSubmit.Caption = "Submit"

    '.cmbCourse.Clear
    .cmbCourse.BackColor = vbWhite

    'Dynamic range based on Support Sheet
    shSupport.Range("A2", shSupport.Range("A" & Rows.Count).End(xlUp)).Name = "Dynamic"

    .cmbCourse.RowSource = "Dynamic"

    .cmbCourse.Value = ""

    .cmbCourse.Value = ""

    'Assigning RowSource to lstDatabase

    .lstDatabase.ColumnCount = 12
    .lstDatabase.ColumnHeads = True

    .lstDatabase.ColumnWidths = "30,70,70,40,45,70,60,60,70,0,0,0"

    iRow = shDatabase.Range("A" & Rows.Count).End(xlUp).row + 1 ' Identify last blank row

    If iRow > 1 Then

        .lstDatabase.RowSource = "Database!A2:L" & iRow

    Else

        .lstDatabase.RowSource = "Database!A2:L2"

    End If


End With
End Sub

On Error GoTo 0

Application.ScreenUpdating = True

 Set oRegEx = CreateObject("VBScript.RegExp")
 With oRegEx
    .Pattern = "^[\w-\.]{1,}\@([\da-zA-Z-]{1,}\.){1,}[\da-zA-Z-]{2,3}$"
    ValidEmail = .Test(Email)
 End With
 Set oRegEx = Nothing

GetImagePath = ""

With Application.FileDialog(msoFileDialogFilePicker) ' File Picker Dialog box

    .AllowMultiSelect = False
    .Filters.Clear      ' Clear the exisiting filters
    .Filters.Add "Images", "*.gif; *.jpg; *.jpeg" 'Add a filter that includes GIF and JPEG images

    ' show the file picker dialog box
    If .Show <> 0 Then

        GetImagePath = .SelectedItems(1) ' Getting the path of selected file name

    End If

End With
End Function
```

```
Sub CreateFolder()
Dim strFolder As String ' To hold the folter path where we need to replicate the image

strFolder = ThisWorkbook.Path & Application.PathSeparator & "Images"
'Check Directory exist or not. If not exist then it will return blank
     If Dir(strFolder, vbDirectory) = "" Then
         MkDir strFolder ' Make a folder with the name of 'Images'
     End If
End Sub



Sub LoadImange()
Dim imgSourcePath As String ' To store the path of image selected by user
Dim imgDestination As String ' To store the path of image selected by user

imgSourcePath = Trim(GetImagePath()) ' Call the Function

If imgSourcePath = "" Then Exit Sub

Call CreateFolder   'Create Image folder if not exist

imgDestination = ThisWorkbook.Path & Application.PathSeparator & _
frmDataEntry.txtStudentName & "." & Split(imgSourcePath, ".")(UBound(Split(imgSourcePath, ".")))

FileCopy imgSourcePath, imgDestination ' Code to copy image

frmDataEntry.imgStudent.PictureSizeMode = fmPictureSizeModeStretch 'Stretch mode
frmDataEntry.imgStudent.Picture = LoadPicture(imgDestination) ' Loading picture to imgStudent
frmDataEntry.txtImagePath.Value = imgDestination ' Assigning the path to text boxFunction ValidEntry()
 As Boolean

ValidEntry = True

With frmDataEntry

    'Default Color

    .txtStudentName.BackColor = vbWhite
    .txtFatherName.BackColor = vbWhite
    .txtDOB.BackColor = vbWhite
    .txtMobile.BackColor = vbWhite
    .txtEmail.BackColor = vbWhite
    .txtAddress.BackColor = vbWhite
    .cmbCourse.BackColor = vbWhite

    'Validating Student Name

    If Trim(.txtStudentName.Value) = "" Then
        MsgBox "Please enter Student's name.", vbOKOnly + vbInformation, "Student Name"
        .txtStudentName.BackColor = vbRed
        .txtStudentName.SetFocus
        ValidEntry = False
        Exit Function
    End If


    'Validating Father's name

    If Trim(.txtFatherName.Value) = "" Then
        MsgBox "Please enter Father's name.", vbOKOnly + vbInformation, "Father Name"
        .txtFatherName.BackColor = vbRed
        .txtFatherName.SetFocus
        ValidEntry = False
        Exit Function
    End If

    'Validating DOB

    If Trim(.txtDOB.Value) = "" Then
        MsgBox "DOB is blank. Please enter DOB.", vbOKOnly + vbInformation, "Invalid Entry"
        .txtDOB.BackColor = vbRed
        ValidEntry = False
        Exit Function
    End If


    'Validating Gender

    If .optFemale.Value = False And .optMale.Value = False Then
        MsgBox "Please select gender.", vbOKOnly + vbInformation, "Invalid Entry"
        ValidEntry = False
        Exit Function
    End If

    'Validating Course

    If Trim(.cmbCourse.Value) = "" Then
        MsgBox "Please select the Course from drop-down.", vbOKOnly + vbInformation, "Course Applied"
        .cmbCourse.BackColor = vbRed
        ValidEntry = False
        Exit Function
    End If

    'Validating Mobile Number

    If Trim(.txtMobile.Value) = "" Or Len(.txtMobile.Value) < 10 Or Not IsNumeric(.txtMobile.Value) Th
en
        MsgBox "Please enter a valid mobile number.", vbOKOnly + vbInformation, "Invalid Entry"
        .txtMobile.BackColor = vbRed
        .txtMobile.SetFocus
        ValidEntry = False
        Exit Function
    End If

    'Validating Email

    If ValidEmail(Trim(.txtEmail.Value)) = False Then
        MsgBox "Please enter a valid email address.", vbOKOnly + vbInformation, "Invalid Entry"
        .txtEmail.BackColor = vbRed
        .txtEmail.SetFocus
        ValidEntry = False
        Exit Function
    End If

    'Validating Address

    If Trim(.txtAddress.Value) = "" Then
        MsgBox "Address is blank. Please enter a valid address.", vbOKOnly + vbInformation, "Invalid E
ntry"
        .txtAddress.BackColor = vbRed
        ValidEntry = False
        Exit Function
    End If

    'Validating Image

    If .imgStudent.Picture Is Nothing Then
       MsgBox "Please upload the PP Size Photo.", vbOKOnly + vbInformation, "Picture"
        ValidEntry = False
        Exit Function
    End If

End With
End Function




Sub Submit_Data()
Dim iRow As Long

If frmDataEntry.txtRowNumber.Value = "" Then

 iRow = shDatabase.Range("A" & Rows.Count).End(xlUp).row + 1 ' Identify last blank row

Else
```

```
    iRow = frmDataEntry.txtRowNumber.Value

End If

With shDatabase.Range("A" & iRow)

.Offset(0, 0).Value = "=Row()-1" 'S. No.

.Offset(0, 1).Value = frmDataEntry.txtStudentName.Value 'Student's Name

.Offset(0, 2).Value = frmDataEntry.txtFatherName.Value    'Father's Name

.Offset(0, 3).Value = frmDataEntry.txtDOB.Value   'DOB

.Offset(0, 4).Value = IIf(frmDataEntry.optFemale.Value = True, "Female", "Male")  'Gender

.Offset(0, 5).Value = frmDataEntry.cmbCourse.Value    'Qualification

.Offset(0, 6).Value = frmDataEntry.txtMobile.Value    'Mobile Number

.Offset(0, 7).Value = frmDataEntry.txtEmail.Value     'Email

.Offset(0, 8).Value = frmDataEntry.txtAddress.Value   'Address

.Offset(0, 9).Value = frmDataEntry.txtImagePath.Value   'Photo

.Offset(0, 10).Value = Application.UserName    'Submitted By

.Offset(0, 11).Value = Format([Now()], "DD-MMM-YYYY HH:MM:SS")   'Submitted On

'Reset the form

Call Reset_Form

Application.ScreenUpdating = True

MsgBox "Data submitted successfully!"
End Sub



Function Selected_List() As Long
Dim i As Long
Selected_List = 0
If frmDataEntry.lstDatabase.ListCount = 1 Then Exit Function ' If no items exist in List Box
For i = 0 To frmDataEntry.lstDatabase.ListCount - 1
If frmDataEntry.lstDatabase.Selected(i) = True Then
   Selected_List = i + 1
   Exit For
End If
Next i
End Function
End Function

Sub Show_Form()
frmDataEntry.Show
End Sub

Private Sub cmdLoadImage_Click()
If Me.txtStudentName.Value = "" Then
 MsgBox "Please enter Student's first.", vbOKOnly + vbCritical, "Error"
 Exit Sub
End If

Call LoadImange
End Sub



Private Sub UserForm6_Initialize()
Call Reset_Form
End Sub
```

```
Private Sub cmdSubmit_Click()
Dim i As VbMsgBoxResult

i = MsgBox("Do you want to submit the data?", vbYesNo + vbQuestion, "Submit Data")

If i = vbNo Then Exit Sub

If ValidEntry Then

    Call Submit_Data

End If
End Sub


Private Sub cmdReset_Click()
Dim i As VbMsgBoxResult

i = MsgBox("Do you want to reset the form?", vbYesNo + vbQuestion, "Reset")

If i = vbNo Then Exit Sub

Call Reset_Form
End Sub


Private Sub lstDatabase_DblClick(ByVal Cancel As MSForms.ReturnBoolean)

If Selected_List = 0 Then
     MsgBox "No row is selected.", vbOKOnly + vbInformation, "Edit"
     Exit Sub
End If

Dim sGender As String

'Me.txtRowNumber = Selected_List + 1 ' Assigning Selected Row Number of Database Sheet

Me.txtRowNumber = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 0) + 1

'Assigning the Selected Reocords to Form controls

frmDataEntry.txtStudentName.Value = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 1)

frmDataEntry.txtFatherName.Value = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 2)

frmDataEntry.txtDOB.Value = Format(Me.lstDatabase.List(Me.lstDatabase.ListIndex, 3), "dd-mmm-yyyy")

sGender = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 4)

If sGender = "Female" Then
    frmDataEntry.optFemale.Value = True
Else
    frmDataEntry.optMale.Value = True
End If

frmDataEntry.cmbCourse.Value = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 5)
frmDataEntry.txtMobile.Value = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 6)
frmDataEntry.txtEmail.Value = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 7)
frmDataEntry.txtAddress.Value = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 8)
frmDataEntry.imgStudent.Picture = LoadPicture(Me.lstDatabase.List(Me.lstDatabase.ListIndex, 9))
frmDataEntry.txtImagePath = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 9)
Me.cmdSubmit.Caption = "Update"
MsgBox "Please make the required changes and Click on Update."
End Sub



Private Sub cmdDelete_Click()
If Selected_List = 0 Then

     MsgBox "No row is selected.", vbOKOnly + vbInformation, "Delete"
     Exit Sub

End If

Dim i As VbMsgBoxResult

Dim row As Long

row = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 0) + 1

i = MsgBox("Do you want ot delete the selected record?", vbYesNo + vbQuestion, "Delete")

If i = vbNo Then Exit Sub

ThisWorkbook.Sheets("Database").Rows(row).Delete

Call Reset ' Refresh the controls with latest information

MsgBox "Selected record has been successfully deleted.", vbOKOnly + vbInformation, "Delete"
End Sub


Private Sub cmdEdit_Click()
If Selected_List = 0 Then

     MsgBox "No row is selected.", vbOKOnly + vbInformation, "Edit"
     Exit Sub

End If

Dim sGender As String

Me.txtRowNumber = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 0) + 1

'Assigning the Selected Reocords to Form controls

frmDataEntry.txtStudentName.Value = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 1)

frmDataEntry.txtFatherName.Value = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 2)

frmDataEntry.txtDOB.Value = Format(Me.lstDatabase.List(Me.lstDatabase.ListIndex, 3), "dd-mmm-yyyy")

sGender = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 4)

If sGender = "Female" Then
    frmDataEntry.optFemale.Value = True
Else
    frmDataEntry.optMale.Value = True
End If
frmDataEntry.cmbCourse.Value = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 5)
frmDataEntry.txtMobile.Value = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 6)
frmDataEntry.txtEmail.Value = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 7)
frmDataEntry.txtAddress.Value = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 8)
frmDataEntry.imgStudent.Picture = LoadPicture(Me.lstDatabase.List(Me.lstDatabase.ListIndex, 9))
frmDataEntry.txtImagePath = Me.lstDatabase.List(Me.lstDatabase.ListIndex, 9)
Me.cmdSubmit.Caption = "Update"
MsgBox "Please make the required changes and Click on Update."
```



student register form

student name

father name

date of birthday

gender ○ O male ○ Optio femele

adress

course applied

mobile number

email adress