

## C# and .NET Frameworks Assignment 2

1. Design and implement a **Student Registration** form using C# and Windows Forms. The form should allow users to input and save student details into a database.

### Requirements:

1. The registration form should include the following fields:
  - **Student ID** (Auto-generated or entered manually)
  - **First Name**
  - **Last Name**
  - **Date of Birth**
  - **Email**
  - **Phone Number**
  - **Course Enrolled**
2. Implement the following features:
  - **Form Validation:** Ensure that all required fields (e.g., Student ID, First Name, Last Name, Email) are properly validated (e.g., email format, required fields, etc.).
  - **Save Data:** Connect the form to a database (SQL Server, MySQL, or any other relational database of your choice) using ADO.NET to save the student data.
  - **Reset Form:** Provide a "Clear" button to reset all the input fields.
  - **Display Students:** Optionally, include a `DataGridView` control to display all registered students after submission.

### Form1

using System;

using System.Windows.Forms;

namespace UserInfoApp

```
{  
public partial class Form1 : Form  
{  
public Form1()  
{  
InitializeComponent();  
}
```

```
private void buttonSubmit_Click(object sender, EventArgs e)  
{  
string name = textBoxName.Text; string address = textBoxAddress.Text;  
string phoneNumber = textBoxPhoneNumber.Text;
```

```
Form2 form2 = new Form2(name, address, phoneNumber); form2.Show();  
}  
}  
}
```

## **Form2**

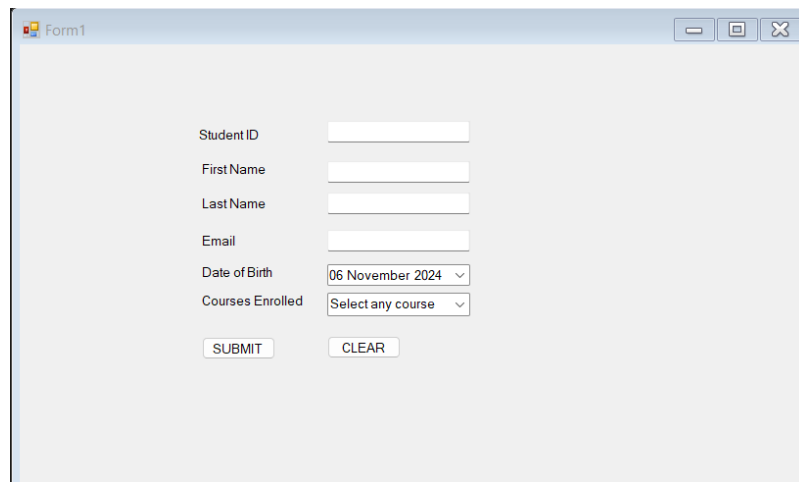
```
using System;
using System.Windows.Forms;

namespace UserInfoApp
{
    public partial class Form2 : Form
    {
        private string _name; private string _address;
        private string _phoneNumber;

        public Form2(string name, string address, string phoneNumber)
        {
            InitializeComponent();
            _name = name;
            _address = address;
            _phoneNumber = phoneNumber;
        }

        private void Form2_Load(object sender, EventArgs e)
        {
            labelDisplayName.Text = "Name: " + _name; labelDisplayAddress.Text = "Address: " +
            _address;
            labelDisplayPhoneNumber.Text = "Phone Number: " + _phoneNumber;
        }
    }
}
```

## **FORM AND CODE:**



The screenshot shows a Windows Form titled "Form1" with a standard Windows XP-style title bar. The form contains a registration form with the following fields and controls:

- Student ID: Text box
- First Name: Text box
- Last Name: Text box
- Email: Text box
- Date of Birth: Date picker (showing 06 November 2024)
- Courses Enrolled: Dropdown menu (showing Select any course)
- SUBMIT: Button
- CLEAR: Button

Form1

Student ID: 73772226157

First Name: Varuna

Last Name: Sree

Email: varu1234@gmail.com

Date of Birth: 27 November 2004

Courses Enrolled: AI&DS

SUBMIT CLEAR

```

1  TextBox txtStudentID, txtFirstName, txtLastName, txtEmail,
    txtPhoneNumber;
2  ComboBox cmbCourseEnrolled;
3  DateTimePicker dtpDateOfBirth;
4  Button btnSubmit, btnClear;
5  DataGridView dgvStudents;
6  private void Form1_Load(object sender, EventArgs e) {
7      LoadStudentData();
8  }
9  private void btnSubmit_Click(object sender, EventArgs e) {
10     if (ValidateForm()) {
11         SaveStudentData();
12         ClearForm();
13         LoadStudentData();
14     }
15 }
16 private bool ValidateForm() {
17     if (string.IsNullOrEmpty(txtFirstName.Text) || string
        .IsNullOrEmpty(txtEmail.Text)) {
18         MessageBox.Show("First Name and Email are required.");
19         return false;
20     }
21     return true;
22 }
23 private void SaveStudentData() {

```

```

24     using (SqlConnection con = new SqlConnection
        ("your_connection_string")) {
25         con.Open();
26         SqlCommand cmd = new SqlCommand("INSERT INTO Students
            (FirstName, LastName, DateOfBirth, Email, PhoneNumber,
            CourseEnrolled) " +
27             "VALUES (@FirstName,
                @LastName, @DateOfBirth, @Email,
                @PhoneNumber, @CourseEnrolled)",
            con);
28         cmd.Parameters.AddWithValue("@FirstName", txtFirstName.Text
            );
29         cmd.Parameters.AddWithValue("@LastName", txtLastName.Text);
30         cmd.Parameters.AddWithValue("@DateOfBirth", dtpDateOfBirth
            .Value);
31         cmd.Parameters.AddWithValue("@Email", txtEmail.Text);
32         cmd.Parameters.AddWithValue("@PhoneNumber", txtPhoneNumber
            .Text);
33         cmd.Parameters.AddWithValue("@CourseEnrolled",
            cmbCourseEnrolled.SelectedItem.ToString());
34         cmd.ExecuteNonQuery();
35     }
36 }
37 private void ClearForm() {
38     txtFirstName.Clear();
39     txtLastName.Clear();

```

2.Design and implement a **Student Fee Payment System** using C# and Windows Forms. The application should allow students to enter their details, pay their fees, and generate a bill with a unique bill number.

### Requirements:

#### 1. Form Design:

- The form should include the following fields:
  - **Student Name**
  - **Roll Number**
  - **Year of Study** (Dropdown or input field)
  - **Hostel/Day Scholar** (Radio buttons or dropdown selection)
  - **Due Date for Fee Payment** (Date Picker)

#### 2. Fee Payment and Bill Generation:

- When the student fills in the required details and clicks the "Pay Fee" button:
  - **Calculate the total fees** based on whether the student is a **Hostel Resident** or a **Day Scholar** (apply appropriate fees based on your assumption).
  - If the payment is made **after the due date**, apply a **late fee of Rs. 100 per day**.
  - Generate a **Bill** with a unique bill number, and display it along with the student's details (name, roll number, year of study, total fees, and any late fees applied).

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Skill Set Selection</title>
<style>
body {
font-family: Arial, sans-serif; background-color: #f4f4f4; display: flex;
justify-content: center; align-items: center; height: 100vh; margin: 0;
}

.form-container { background-color: white; padding: 20px;
border-radius: 5px;
box-shadow: 0 0 10px rgba(0, 0, 0, 0.1); width: 300px;
}

h2 {
text-align: center; margin-bottom: 20px;
}

.checkbox-container { margin-bottom: 15px;
}

.checkbox-container label { margin-left: 10px;
}
}
```

```
.checkbox-container input[type="checkbox"] { transform: scale(1.2);
margin-right: 10px;
}
button {
width: 100%; padding: 10px;
background-color: #007BFF;
color: white; border: none; border-radius: 5px; cursor: pointer;
}
```

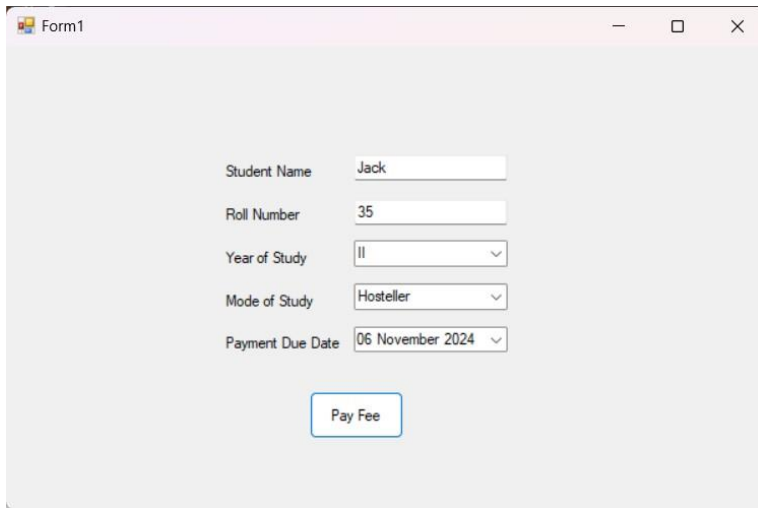
```
button:hover {
background-color: #0056b3;
}
```

```
</style>
</head>
<body>
```

```
<div class="form-container">
<h2>Select Your Skill Set</h2>
<form>
<div class="checkbox-container">
<input type="checkbox" id="skill1" name="skill" value="HTML">
<label for="skill1">HTML</label>
</div>
<div class="checkbox-container">
<input type="checkbox" id="skill2" name="skill" value="CSS">
<label for="skill2">CSS</label>
</div>
<div class="checkbox-container">
<input type="checkbox" id="skill3" name="skill" value="JavaScript">
<label for="skill3">JavaScript</label>
</div>
<div class="checkbox-container">
<input type="checkbox" id="skill4" name="skill" value="Python">
<label for="skill4">Python</label>
</div>
<button type="submit">Submit</button>
</form>
</div>

</body>
</html>
```

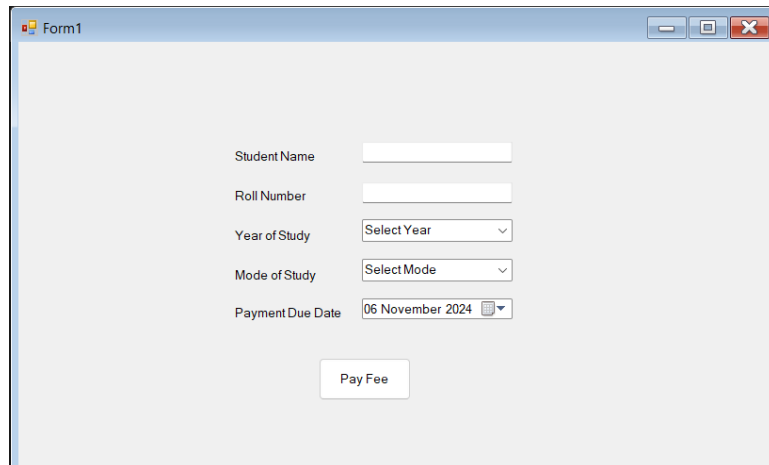
## FORM AND CODE:



A screenshot of a Windows application window titled "Form1". The window has a light gray background and a standard Windows title bar with minimize, maximize, and close buttons. The form contains five labeled text boxes and a button. The data entered in the text boxes is: Student Name: Jack, Roll Number: 35, Year of Study: II, Mode of Study: Hosteller, and Payment Due Date: 06 November 2024. The "Pay Fee" button is located below the text boxes.

Student Name	Jack
Roll Number	35
Year of Study	II
Mode of Study	Hosteller
Payment Due Date	06 November 2024

Pay Fee



A screenshot of a Windows application window titled "Form1". The window has a light gray background and a standard Windows title bar with minimize, maximize, and close buttons. The form contains five labeled text boxes and a button. The data entered in the text boxes is: Student Name: (empty), Roll Number: (empty), Year of Study: Select Year, Mode of Study: Select Mode, and Payment Due Date: 06 November 2024. The "Pay Fee" button is located below the text boxes.

Student Name	
Roll Number	
Year of Study	Select Year
Mode of Study	Select Mode
Payment Due Date	06 November 2024

Pay Fee

```

1 TextBox txtStudentName, txtRollNumber;
2 ComboBox cmbYearOfStudy, cmbHostelDayScholar;
3 DateTimePicker dtpDueDate;
4 Button btnPayFee;
5 Label lblBill;
6 const decimal HostelFee = 50000;
7 const decimal DayScholarFee = 30000;
8 const decimal LateFeePerDay = 100;
9 private void btnPayFee_Click(object sender, EventArgs e) {
10     if (ValidateForm()) {
11         decimal totalFee = CalculateFee();
12         string billNumber = GenerateBillNumber();
13         DisplayBill(billNumber, totalFee);
14     }
15 }
16 private bool ValidateForm() {
17     if (string.IsNullOrEmpty(txtStudentName.Text) || string.IsNullOrEmpty(
18         txtRollNumber.Text)) {
19         MessageBox.Show("Student Name and Roll Number are required.");
20         return false;
21     }
22     return true;
23 }
24 private decimal CalculateFee()
25 {
26     decimal baseFee;
27     if (cmbHostelDayScholar.SelectedItem.ToString() == "Hostel Resident") {
28         baseFee = HostelFee;
29     }
30     else {
31         baseFee = DayScholarFee;
32     }
33     DateTime dueDate = dtpDueDate.Value;
34     DateTime paymentDate = DateTime.Today;
35     decimal lateFee = 0;
36     if (paymentDate > dueDate) {
37         int daysLate = (paymentDate - dueDate).Days;
38         lateFee = daysLate * LateFeePerDay;
39     }
40     return baseFee + lateFee;
41 }
42 private string GenerateBillNumber() {
43     return "BILL" + DateTime.Now.Ticks.ToString() + txtRollNumber.Text;
44 }
45 private void DisplayBill(string billNumber, decimal totalFee) {
46     lblBill.Text = $"Bill Number: {billNumber}\n" +
47         $"Student Name: {txtStudentName.Text}\n" +
48         $"Roll Number: {txtRollNumber.Text}\n" +
49         $"Year of Study: {cmbYearOfStudy.SelectedItem.ToString()}\n" +
50         $"Hostel/Day Scholar: {cmbHostelDayScholar.SelectedItem
51         .ToString()}\n" +
52         $"Total Fees: {totalFee:C}";
53 }

```

3. Design and implement a **Web Service** using C# and ASP.NET to expose functionality for client applications to consume. The web service should provide a specific set of operations, such as retrieving data or performing a calculation.

### **Implementation Steps:**

#### **Define the Service Requirements:**

- Identify the functionality that the web service will provide (e.g., retrieving student information, performing fee calculations, etc.).
- Specify the input parameters and return types for each operation that the web service will expose.

#### **Create the Web Service:**

- In Visual Studio, create a new **ASP.NET Web Service** project.
- Define a service class by inheriting from `System.Web.Services.WebService`.
- Decorate the class with the `[WebService]` attribute and each method with the `[WebMethod]` attribute to expose them as web service operations.
- Implement the required service methods (e.g., retrieving student data or calculating fees).

```
using System;
```

```
using System.Windows.Forms;
```

```
namespace StudentFeePaymentSystem
```

```
{
```

```
public partial class Form1 : Form
```

```
{
```

```
private static int billCounter = 1000; // Starting bill number
```

```
public Form1()
```

```
{
```

```
InitializeComponent();
```

```
}
```

```
private void buttonPayFee_Click(object sender, EventArgs e)
```

```
{
```

```
// Get student details
```

```
string name = textBoxName.Text;
```

```
string rollNumber = textBoxRollNumber.Text;
```

```
int yearOfStudy = int.Parse(comboBoxYear.SelectedItem.ToString()); bool isHosteller =
```

```
radioButtonHosteller.Checked;
```

```
DateTime dueDate = dateTimePickerDueDate.Value; DateTime paymentDate =
```

```
DateTime.Now;
```

```
// Base fees
```

```
int baseFee = isHosteller ? 50000 : 30000;
```

```
// Late fee calculation int lateFee = 0;
```



```

if (paymentDate > dueDate)
{
lateFee = (paymentDate - dueDate).Days * 100;
}

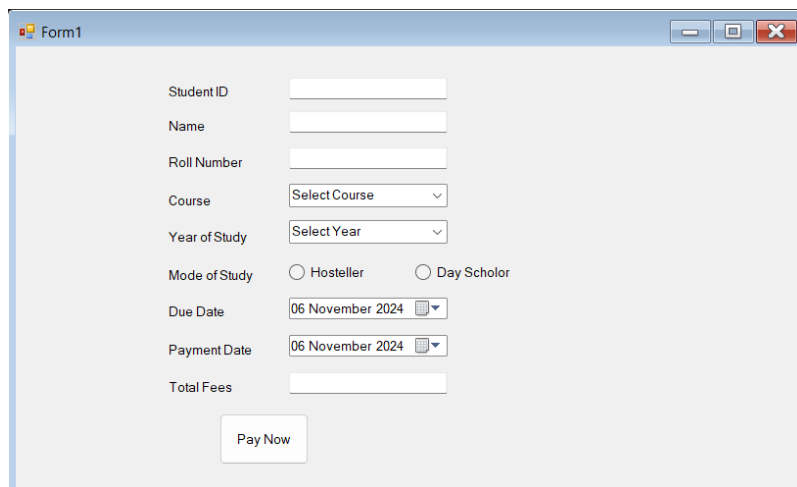
// Total fee
int totalFee = baseFee + lateFee;

// Generate bill number
int billNumber = GenerateBillNumber();

// Display the bill
textBoxBill.Text = GenerateBill(name, rollNumber, yearOfStudy, isHosteller, paymentDate,
totalFee, billNumber);
}
private int GenerateBillNumber()
{
return billCounter++;
}
private string GenerateBill(string name, string rollNumber, int yearOfStudy, bool isHosteller,
DateTime paymentDate, int totalFee, int billNumber)
{
string studentType = isHosteller ? "Hosteller" : "Day Scholar"; return $"Bill Number:
{billNumber}\n" +
$"Name: {name}\n" +
$"Roll Number: {rollNumber}\n" +
$"Year of Study: {yearOfStudy}\n" +
$"Student Type: {studentType}\n" +
$"Payment Date: {paymentDate.ToShortDateString()}\n" +
$"Total Fee: Rs. {totalFee}";
}
}
}

```

### FORM AND CODE:



The screenshot shows a Windows application window titled "Form1". Inside the window, there is a form with the following fields and controls:

- Student ID: Text input field
- Name: Text input field
- Roll Number: Text input field
- Course: Dropdown menu with "Select Course" as the placeholder
- Year of Study: Dropdown menu with "Select Year" as the placeholder
- Mode of Study: Two radio buttons labeled "Hosteller" and "Day Scholor" (note the typo in the image)
- Due Date: Date picker showing "06 November 2024"
- Payment Date: Date picker showing "06 November 2024"
- Total Fees: Text input field
- Pay Now: Button

Form1

Student ID

Name

Roll Number

Course

Year of Study

Mode of Study ☐ Hosteller ☒ Day Scholor

Due Date

Payment Date

Total Fees

```

1 using System;
2 using System.Web.Services;
3 [WebService(Namespace = "http://yourdomain.com/")]
4 [WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
5 public class StudentService : WebService {
6     private static readonly Dictionary<int, Student> students = new Dictionary<int
7         , Student> {
8         { 1, new Student { StudentID = 1, Name = "John Doe", RollNumber = "1001",
9             Course = "Computer Science", YearOfStudy = 2 } },
10        { 2, new Student { StudentID = 2, Name = "Jane Smith", RollNumber = "1002"
11            , Course = "Electrical Engineering", YearOfStudy = 3 } }
12    };
13    public Student GetStudentDetails(int studentID) {
14        if (students.ContainsKey(studentID)) {
15            return students[studentID];
16        }
17        else {
18            return null;
19        }
20    }
21    public decimal CalculateFees(int studentID, bool isHostelResident, DateTime
22        dueDate, DateTime paymentDate) {
23        decimal baseFee = isHostelResident ? 50000 : 30000;
24        decimal lateFee = 0;
25        if (paymentDate > dueDate) {
26            int daysLate = (paymentDate - dueDate).Days;
27            lateFee = daysLate * 100;
28        }
29        return baseFee + lateFee;
30    }
31 }
32 public class Student {
33     public int StudentID { get; set; }
34     public string Name { get; set; }
35     public string RollNumber { get; set; }
36     public string Course { get; set; }
37     public int YearOfStudy { get; set; }
38 }

```

4. Our college is organizing an **Alumni Meet** on **May 5, 2024**. The alumni cell is in the process of creating a database to store a list of registered alumni who will attend the event. You are tasked with designing a registration form and implementing it using ADO.NET.

### Requirements:

#### 1. Design the Registration Form:

- Create a Windows Forms application that includes the following controls:
  - **TextBox** for entering the **Alumni Name**
  - **TextBox** for entering the **Email**
  - **TextBox** for entering the **Phone Number**
  - **ComboBox** for selecting the **Department** (e.g., Computer Science, Business, Arts)
  - **Button** to **Register** alumni
  - **Button** to **Display** registered alumni
  - **DataGridView** control to display the list of registered alumni from the selected department

#### 2. Implement Functionality Using ADO.NET:

- **Register Button:**
  - When the **Register** button is clicked, validate the input fields.
  - If the inputs are valid, insert the entered details into the database using ADO.NET. Handle any database exceptions that may occur.
- **Display Button:**
  - When the **Display** button is clicked, retrieve all registered alumni for the selected department from the ComboBox.
  - Display the results in the **DataGridView** control.

### ASP Design

```
<asp:Label ID="lblName" runat="server" Text="Name:"></asp:Label>
<asp:TextBox ID="txtName" runat="server"></asp:TextBox><br />
```

```
<asp:Label ID="lblRollNumber" runat="server" Text="Roll Number:"></asp:Label>
<asp:TextBox ID="txtRollNumber" runat="server"></asp:TextBox><br />
```

```
<asp:Label ID="lblYear" runat="server" Text="Year of Study:"></asp:Label>
<asp:DropDownList ID="ddlYear" runat="server">
<asp:ListItem Text="1" Value="1"></asp:ListItem>
<asp:ListItem Text="2" Value="2"></asp:ListItem>
<asp:ListItem Text="3" Value="3"></asp:ListItem>
<asp:ListItem Text="4" Value="4"></asp:ListItem>
</asp:DropDownList><br />
```

```
<asp:Label ID="lblBranch" runat="server" Text="Branch:"></asp:Label>
<asp:TextBox ID="txtBranch" runat="server"></asp:TextBox><br />
```

```
<asp:Label ID="lblCGPA" runat="server" Text="CGPA:"></asp:Label>
<asp:TextBox ID="txtCGPA" runat="server"></asp:TextBox><br />
```

```
<asp:Button ID="btnRegister" runat="server" Text="Register" OnClick="btnRegister_Click" />
<asp:Button ID="btnDisplay" runat="server" Text="Display" OnClick="btnDisplay_Click" /><br />
```

```
<asp:GridView ID="gvStudents" runat="server"></asp:GridView>
```

### **Register Click**

```
using System; using System.Data;
using System.Data.SqlClient; using System.Configuration;
protected void btnRegister_Click(object sender, EventArgs e)
{
    string connectionString =
    ConfigurationManager.ConnectionStrings["PlacementDBConnectionString"].Connection
    String;

    using (SqlConnection con = new SqlConnection(connectionString))
    {
        string query = "INSERT INTO Students (Name, RollNumber, Year, Branch, CGPA) VALUES
        (@Name, @RollNumber, @Year, @Branch, @CGPA)";
        using (SqlCommand cmd = new SqlCommand(query, con))
        {
            cmd.Parameters.AddWithValue("@Name", txtName.Text);
            cmd.Parameters.AddWithValue("@RollNumber", txtRollNumber.Text);
            cmd.Parameters.AddWithValue("@Year", ddlYear.SelectedValue);
            cmd.Parameters.AddWithValue("@Branch", txtBranch.Text);
            cmd.Parameters.AddWithValue("@CGPA",
            Convert.ToDouble(txtCGPA.Text));

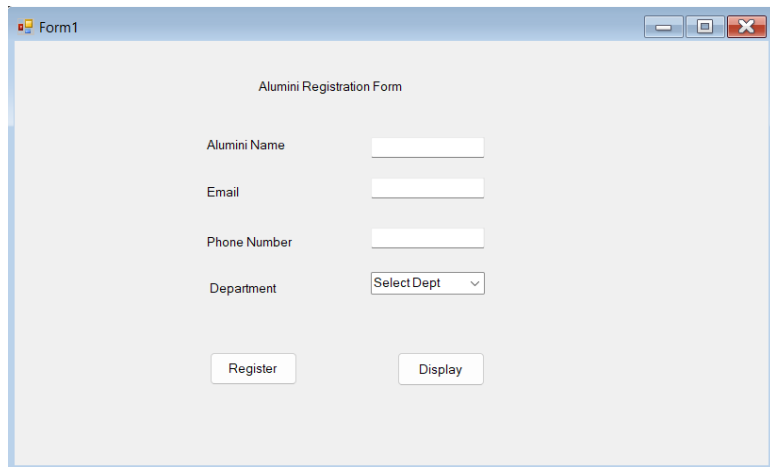
            con.Open(); cmd.ExecuteNonQuery(); con.Close();
        }
    }
}
```

### **Display**

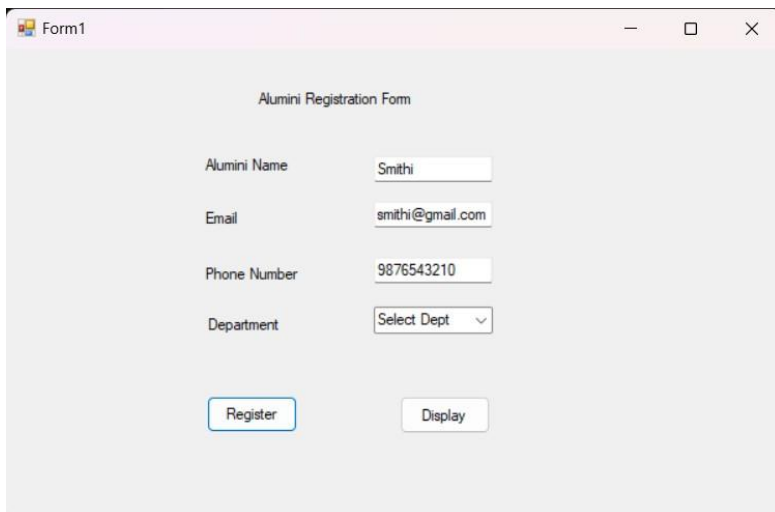
```
protected void btnDisplay_Click(object sender, EventArgs e)
{
    string connectionString =
    ConfigurationManager.ConnectionStrings["PlacementDBConnectionString"].Connection
    String;

    using (SqlConnection con = new SqlConnection(connectionString))
    {
        string query = "SELECT * FROM Students";
        using (SqlDataAdapter sda = new SqlDataAdapter(query, con))
        {
            DataTable dt = new DataTable(); sda.Fill(dt); gvStudents.DataSource = dt;
            gvStudents.DataBind();
        }
    }
}
```

## FORM AND CODE:



A screenshot of a Windows-style application window titled "Form1". The window contains a form titled "Alumini Registration Form". The form has four input fields: "Alumini Name", "Email", "Phone Number", and "Department". The "Department" field is a dropdown menu with "Select Dept" and a downward arrow. Below the input fields are two buttons: "Register" and "Display".



A screenshot of the same "Form1" window, but now with sample data entered into the input fields. The "Alumini Name" field contains "Smithi", the "Email" field contains "smithi@gmail.com", and the "Phone Number" field contains "9876543210". The "Department" dropdown menu still shows "Select Dept". The "Register" and "Display" buttons are still present.

```

1- using System;
2- using System.Data.SqlClient;
3- using System.Windows.Forms;
4- public partial class Form1 : Form {
5-     string connectionString = @"Data Source=YourServer;Initial
        Catalog=YourDatabase;Integrated Security=True;";
6-     public Form1() {
7-         InitializeComponent();
8-     }
9-     private void btnRegister_Click(object sender, EventArgs e) {
10-         if (ValidateForm()) {
11-             try {
12-                 using (SqlConnection con = new SqlConnection
                    (connectionString)) {
13-                     con.Open();
14-                     string query = "INSERT INTO Alumni (AlumniName,
                        Email, PhoneNumber, Department) " +
15-                         "VALUES (@AlumniName, @Email,
                            @PhoneNumber, @Department)";
16-
17-                     SqlCommand cmd = new SqlCommand(query, con);
18-                     cmd.Parameters.AddWithValue("@AlumniName",
                        txtAlumniName.Text);
19-                     cmd.Parameters.AddWithValue("@Email", txtEmail
                        .Text);

```

```

20-                     cmd.Parameters.AddWithValue("@PhoneNumber",
                        txtPhoneNumber.Text);
21-                     cmd.Parameters.AddWithValue("@Department",
                        cmbDepartment.SelectedItem.ToString());
22-
23-                     cmd.ExecuteNonQuery();
24-                     MessageBox.Show("Alumni registered
                        successfully!");
25-                     ClearForm();
26-                 }
27-             }
28-             catch (Exception ex) {
29-                 MessageBox.Show("Error: " + ex.Message);
30-             }
31-         }
32-     }
33-     private bool ValidateForm() {
34-         if (string.IsNullOrEmpty(txtAlumniName.Text) || string
            .IsNullOrEmpty(txtEmail.Text) ||
35-             string.IsNullOrEmpty(txtPhoneNumber.Text) ||
            cmbDepartment.SelectedItem == null) {
36-             MessageBox.Show("All fields are required!");
37-             return false;
38-         }
39-         return true;
40-     }

```

```

41-     private void ClearForm() {
42-         txtAlumniName.Clear();
43-         txtEmail.Clear();
44-         txtPhoneNumber.Clear();
45-         cmbDepartment.SelectedIndex = -1;
46-     }
47- }

```

**BY:**

VENKATESHWARAN S

73772226159

III – B.TECH AI&DS

