

C# and .NET Frameworks

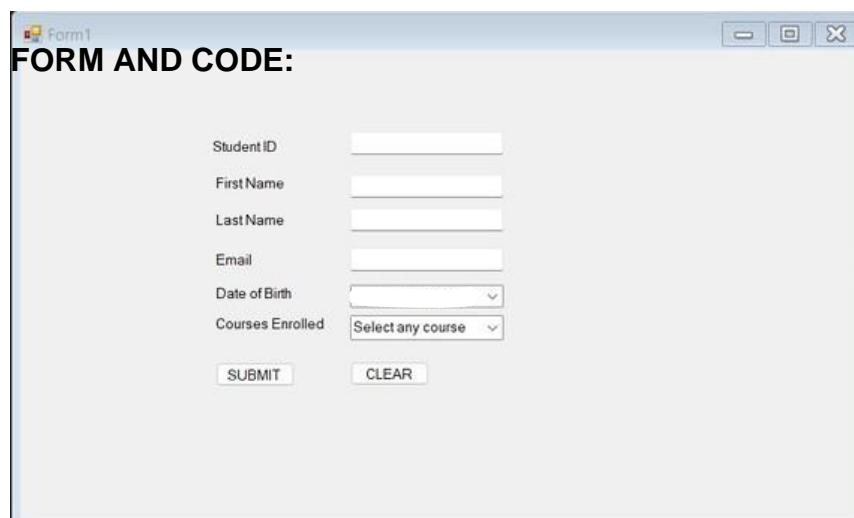
Assignment 2

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:
 - o **Student ID** (Auto-generated or entered manually)
 - o **First Name**
 - o **Last Name**
 - o **Date of Birth**
 - o **Email**
 - o **Phone Number**
 - o **Course Enrolled**
2. Implement the following features:
 - o **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.).
 - o **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.
 - o **Reset Form:** Provide a "Clear" button to reset all the input fields.
 - o **Display Students:** Optionally, include a DataGridView control to display all registered students after submission.

FORM AND CODE:



Student ID

First Name

Last Name

Email

Date of Birth

Courses Enrolled

Form1

Student ID

First Name

Last Name

Email

Date of Birth

Courses Enrolled

```

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:

o The form should include the following fields:

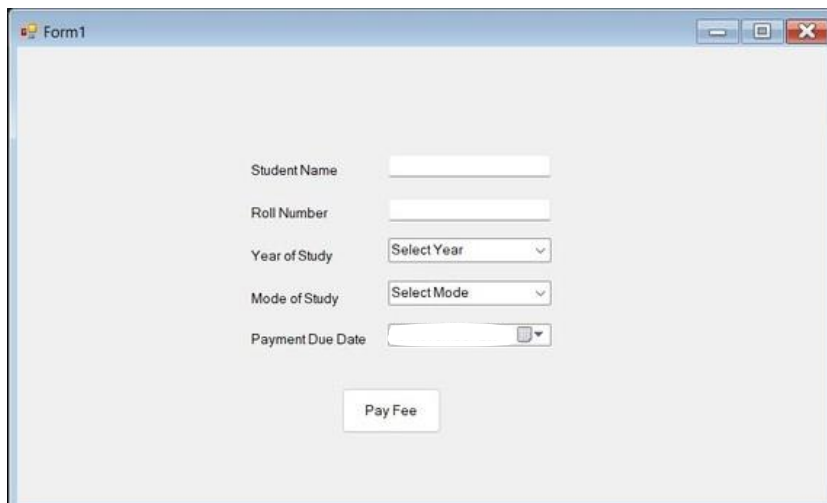
- ☐ **Student Name Roll Number Year of Study**
- ☐ (Dropdown or input field) **Hostel/Day Scholar** (Radio
- ☐ buttons or dropdown selection) **Due Date for Fee**
- ☐ **Payment** (Date Picker)

2. Fee Payment and Bill Generation:

o 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).

FORM AND CODE:



The screenshot shows a Windows Form titled "Form1" with a light gray background. It contains five input fields arranged vertically on the left, each with a label and a corresponding control on the right:

- Student Name**: A text input field.
- Roll Number**: A text input field.
- Year of Study**: A dropdown menu with "Select Year" as the placeholder text.
- Mode of Study**: A dropdown menu with "Select Mode" as the placeholder text.
- Payment Due Date**: A date picker control showing a calendar icon.

Below these fields, centered, is a button labeled "Pay Fee".

Form1

Student Name: Jack

Roll Number: 35

Year of Study: II

Mode of Study: Hosteller

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.ToString()}\n" +
51         $"Total Fees: {totalFee:C}";

```

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:

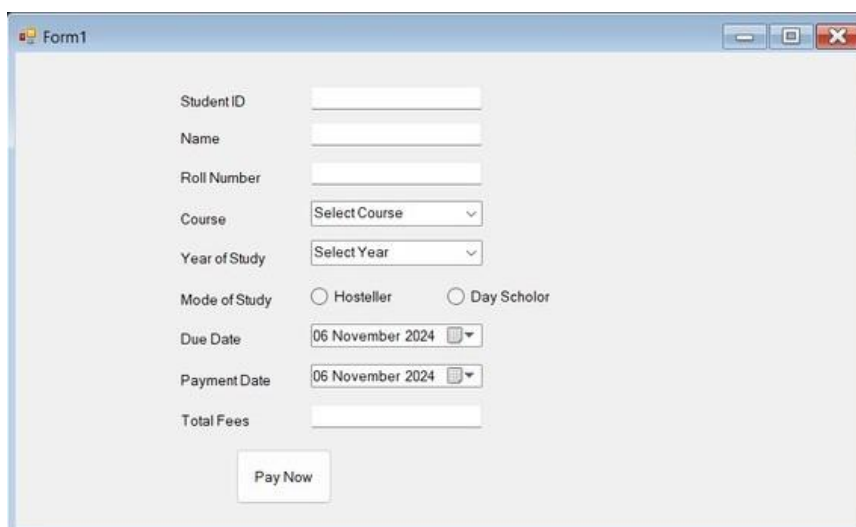
Define the Service Requirements:

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

Create the Web Service:

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

FORM AND CODE:



The screenshot shows a web form titled "Form1" 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" text
- Year of Study: Dropdown menu with "Select Year" text
- Mode of Study: Two radio buttons labeled "Hosteller" and "Day Scholor"
- 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: 1

Name: John Doe

Roll Number: 1001

Course: AI&DS

Year of Study: IV

Mode of Study: ☐ Hosteller ☒ Day Scholar

Due Date: 12 November 2024

Payment Date: 08 November 2024

Total Fees: 60000

Pay Now

```

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

```

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:

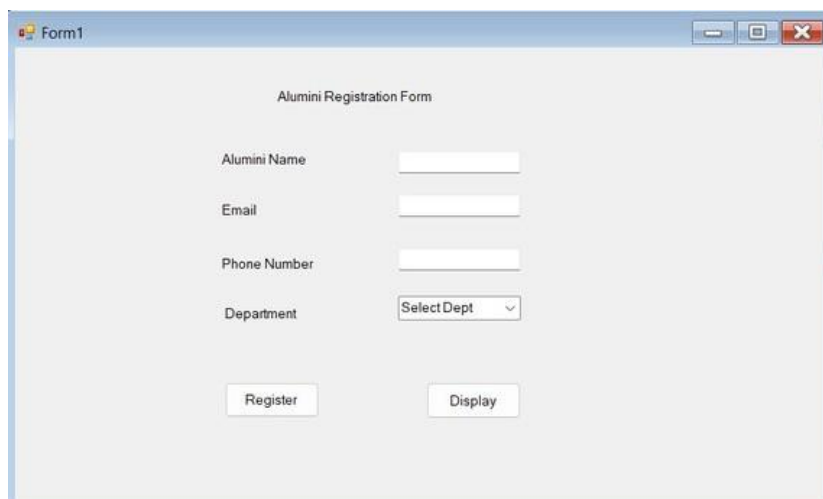
1. Design the Registration Form:

- o 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:

- o **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.
- o **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.

FORM AND CODE:



The screenshot shows a Windows Forms application window titled "Form1". Inside the window, there is a form titled "Alumini Registration Form". The form contains the following controls:

- Labels: "Alumini Name", "Email", "Phone Number", "Department".
- Input fields: Three text boxes for "Alumini Name", "Email", and "Phone Number".
- ComboBox: A dropdown menu for "Department" with the text "Select Dept" and a downward arrow.
- Buttons: Two buttons labeled "Register" and "Display".

Form1

Alumini Registration Form

Alumini Name

Email

Phone Number

Department

```

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-
21-                     cmd.Parameters.AddWithValue("@PhoneNumber",
                        txtPhoneNumber.Text);
22-                     cmd.Parameters.AddWithValue("@Department",
                        cmbDepartment.SelectedItem.ToString());
23-
24-                     cmd.ExecuteNonQuery();
25-                     MessageBox.Show("Alumni registered
                        successfully!");
26-                     ClearForm();
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:

VARUNA SREE N

73772226157

III - B.TECH AI&DS