C# and .NET

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.

Aim:

To design and implement a Student Registration Form using C# and Windows Forms, allowing users to input student details and save them to a database.

Program:

SQL Table

```
CREATE TABLE Students (
    StudentID INT IDENTITY(1,1) PRIMARY KEY,
    FirstName NVARCHAR(50),
```

```
    LastName NVARCHAR(50),
    DateOfBirth DATE,
    Email NVARCHAR(100),
    PhoneNumber NVARCHAR(20),
    CourseEnrolled NVARCHAR(100)
);
```

Form Fields Validation

```
private bool ValidateForm()
{
    bool isValid = true;
        if (string.IsNullOrWhiteSpace(txtFirstName.Text))
    {
        errorProvider1.SetError(txtFirstName, "First Name is required");
        isValid = false;
    }
    if (string.IsNullOrWhiteSpace(txtLastName.Text))
    {
        errorProvider1.SetError(txtLastName, "Last Name is required");
        isValid = false;
    }
    if (!IsValidEmail(txtEmail.Text))
    {
        errorProvider1.SetError(txtEmail, "Invalid email format");
        isValid = false;
    }
    return isValid;
}
private bool IsValidEmail(string email)
```

```
{
    try
    {
        var mail = new System.Net.Mail.MailAddress(email);
        return mail.Address == email;
    }
    catch
    {
        return false;
    }
}
```

Database Connection and Saving Data

```
using System.Data.SqlClient;
private void SaveStudentData()
{
    if (!ValidateForm()) return;
    string connectionString = "your_connection_string_here";
    using (SqlConnection conn = new SqlConnection(connectionString))
    {
        conn.Open();
        string query = "INSERT INTO Students (FirstName, LastName, DateOfBirth,
Email, PhoneNumber, CourseEnrolled) " +
                       "VALUES (@FirstName, @LastName, @DateOfBirth, @Email,
@PhoneNumber, @CourseEnrolled)";
        using (SqlCommand cmd = new SqlCommand(query, conn))
        {
            cmd.Parameters.AddWithValue("@FirstName", txtFirstName.Text);
```

```
            cmd.Parameters.AddWithValue("@LastName", txtLastName.Text);
            cmd.Parameters.AddWithValue("@DateOfBirth", dtpDateOfBirth.Value);
            cmd.Parameters.AddWithValue("@Email", txtEmail.Text);
            cmd.Parameters.AddWithValue("@PhoneNumber", txtPhoneNumber.Text);
            cmd.Parameters.AddWithValue("@CourseEnrolled", txtCourseEnrolled.Text);
            cmd.ExecuteNonQuery();
        }
    }
    MessageBox.Show("Student data saved successfully!");
    ClearForm();
}
```

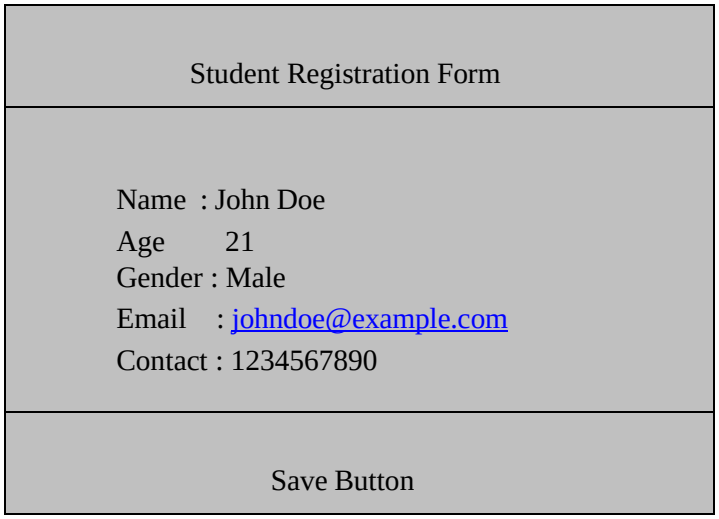
Clear Form Fields

```
private void ClearForm()
{
    txtFirstName.Clear();
    txtLastName.Clear();
    txtEmail.Clear();
    txtPhoneNumber.Clear();
    txtCourseEnrolled.Clear();
    dtpDateOfBirth.Value = DateTime.Now;
    errorProvider1.Clear();
}
private void btnClear_Click(object sender, EventArgs e)
{
    ClearForm();
}
```

Display Students in DataGridView

```
private void LoadStudents()
{
    string connectionString = "your_connection_string_here";
    using (SqlConnection conn = new SqlConnection(connectionString))
    {
        conn.Open();
        string query = "SELECT * FROM Students";
        using (SqlCommand cmd = new SqlCommand(query, conn))
        {
            SqlDataAdapter adapter = new SqlDataAdapter(cmd);
            DataTable dt = new DataTable();
            adapter.Fill(dt);
            dataGridView1.DataSource = dt;
        }
    }
}
```

Output:



"Student registered successfully."

Student ID	First Name	Last Name	Date of Birth	Email	Phone Number	Course Enrolled
1	John	Doe	1999-10-12	john.doe@gmail.com	1234567890	Computer Science
2	Jane	Smith	2000-05-15	jane.smith@yahoo.com	0987654321	Mathematics

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).

Aim:

To design and implement a Student Fee Payment System using C# and Windows Forms, allowingstudents to enter their details, pay their fees, and generate a bill with a unique bill number.

Program:

Form Fields and Input Validation

```
private bool ValidateForm()
{
    bool isValid = true;
    if (string.IsNullOrWhiteSpace(txtStudentName.Text))
    {
        errorProvider1.SetError(txtStudentName, "Student Name is required");
        isValid = false;
    }
    if (string.IsNullOrWhiteSpace(txtRollNumber.Text))
    {
        errorProvider1.SetError(txtRollNumber, "Roll Number is required");
        isValid = false;
    }
    if (cmbYearOfStudy.SelectedIndex == -1)
    {
        errorProvider1.SetError(cmbYearOfStudy, "Year of Study is required");
        isValid = false;
    }
    return isValid;
}
```

Fee Calculation Logic

```
private void CalculateFees()
{
    if (!ValidateForm()) return;
    decimal baseFee = 0;
    decimal lateFee = 0;
    decimal totalFee = 0;
    DateTime dueDate = dtpDueDate.Value;
    DateTime paymentDate = DateTime.Now;
    int daysLate = (paymentDate - dueDate).Days;
    if (rbtnHostel.Checked)
    {
        baseFee = 50000;
    }
    else if (rbtnDayScholar.Checked)
    {
        baseFee = 30000;
    }
    if (daysLate > 0)
    {
        lateFee = daysLate * 100;
    }
    totalFee = baseFee + lateFee;
    string billNumber = GenerateBillNumber();
    txtBillNumber.Text = billNumber;
    txtTotalFee.Text = totalFee.ToString("C");
    txtLateFee.Text = lateFee.ToString("C");
}
```

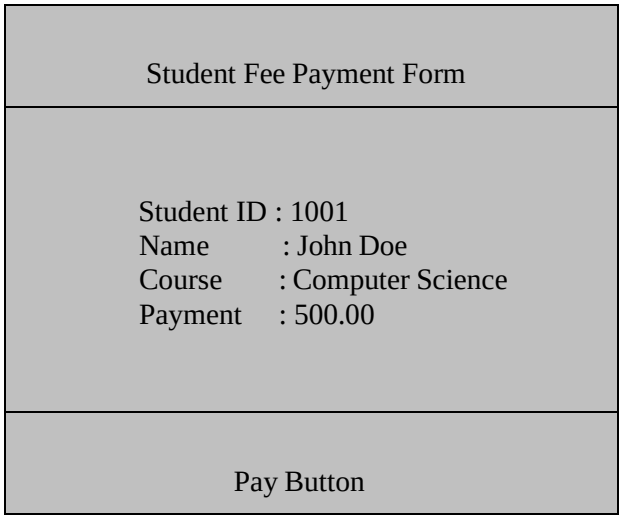
Generating a Unique Bill Number

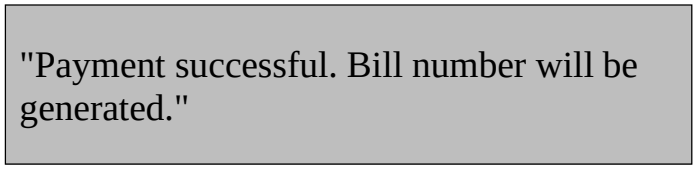
```
private string GenerateBillNumber()
{
   return "BILL-" + DateTime.Now.Ticks.ToString();
}
```

Handling the "Pay Fee" Button

```
private void btnPayFee_Click(object sender, EventArgs e)
{
CalculateFees();
}
```

Output:





Bill Generated Successfully!

Bill Number : 8a5d9c2e-2c43...
Student ID : 1001
Name : John Doe
Course : Computer Science
Amount Paid : $500.00
Date : [Payment Date]

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).

Aim:

To design and implement a Web Service using C# and ASP.NET that exposes specific functionality for clientapplications to consume, such as retrieving data or performing calculations.

Program:

```
using System;
using System.Web.Services;
namespace StudentFeeWebService
{
    [WebService(Namespace = "http://localhost/StudentFeeService")]
    [WebServiceBinding(ConformsTo = WsiProfiles.BasicProfile1_1)]
    [System.ComponentModel.ToolboxItem(false)]
    public class StudentFeeService : WebService
    {
        private static readonly Dictionary<string, Student> students = new
Dictionary<string, Student>
        {
            { "1", new Student { StudentId = "1", Name = "John Doe", RollNumber =
"1001", YearOfStudy = "3rd Year", IsHostelResident = true } },
            { "2", new Student { StudentId = "2", Name = "Jane Smith", RollNumber =
"1002", YearOfStudy = "2nd Year", IsHostelResident = false } }
        };
        [WebMethod(Description = "Retrieve student information by student ID")]
        public Student GetStudentInfo(string studentId)
        {
            if (students.ContainsKey(studentId))
            {
                return students[studentId];
            }
            else
            {
                throw new Exception("Student not found");
            }
        }
```

```
    [WebMethod(Description = "Calculate fee for a student, including late fees if applicable")]
    public FeeDetails CalculateFee(string studentId, DateTime dueDate)
    {
        Student student = GetStudentInfo(studentId);
        decimal baseFee = student.IsHostelResident ? 50000 : 30000;
        decimal lateFee = 0;
        DateTime currentDate = DateTime.Now;
        int daysLate = (currentDate - dueDate).Days;
        if (daysLate > 0)
        {
            lateFee = daysLate * 100; // Rs. 100 per day late fee
        }
        decimal totalFee = baseFee + lateFee;
        return new FeeDetails
        {
            StudentId = studentId,
            BaseFee = baseFee,
            LateFee = lateFee,
            TotalFee = totalFee
        };
    }
}
public class Student
{
    public string StudentId { get; set; }
    public string Name { get; set; }
    public string RollNumber { get; set; }
    public string YearOfStudy { get; set; }
```

```
        public bool IsHostelResident { get; set; }
    }
    public class FeeDetails
    {
        public string StudentId { get; set; }
        public decimal BaseFee { get; set; }
        public decimal LateFee { get; set; }
        public decimal TotalFee { get; set; }
    }
}
```

```
<Student>
    <StudentId>1</StudentId>
    <Name>John Doe</Name>
    <RollNumber>1001</RollNumber>
    <YearOfStudy>3rd Year</YearOfStudy>
    <IsHostelResident>true</IsHostelResident>
</Student>
```

```
<FeeDetails>
    <StudentId>1</StudentId>
    <BaseFee>50000</BaseFee>
    <LateFee>3500</LateFee>
    <TotalFee>53500</TotalFee>
</FeeDetails>
```

Input:

studentId = "1"

dueDate = "2024-10-01"

Output:

```
<Student>
   <StudentId>1</StudentId>
   <Name>John Doe</Name>
   <RollNumber>1001</RollNumber>
   <YearOfStudy>3rd Year</YearOfStudy>
   <IsHostelResident>true</IsHostelResident>
</Student>

<FeeDetails>
   <StudentId>1</StudentId>
   <BaseFee>50000</BaseFee>
   <LateFee>3500</LateFee>
   <TotalFee>53500</TotalFee>
</FeeDetails>
```

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.

Aim:

To design a Windows Forms application for alumni registration for the Alumni Meet and implementfunctionality to store and display registered alumni details using ADO.NET.

Program:

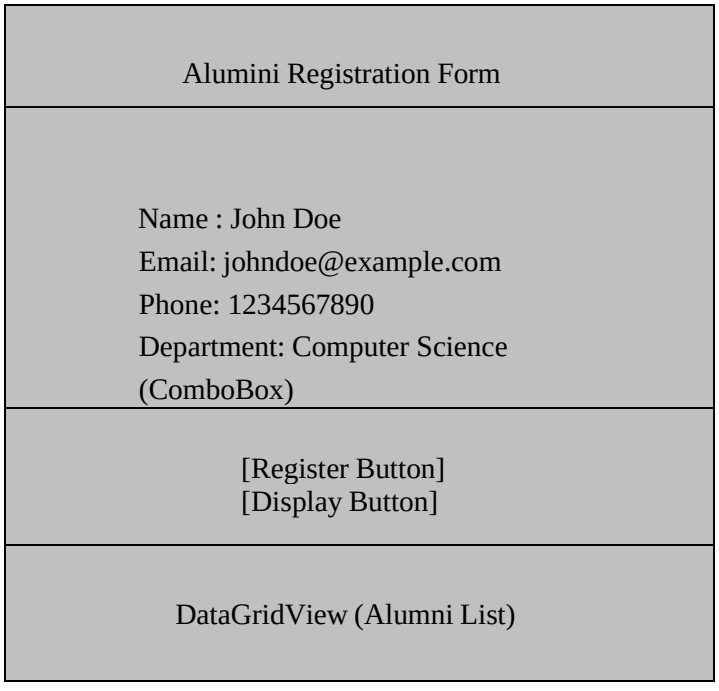
```
CREATE TABLE Alumni (
   AlumniId INT IDENTITY(1,1) PRIMARY KEY,
   AlumniName NVARCHAR(100),
   Email NVARCHAR(100),
   PhoneNumber NVARCHAR(15),
   Department NVARCHAR(50)
);


private void btnRegister_Click(object sender, EventArgs e)
{
   if (string.IsNullOrEmpty(txtAlumniName.Text) ||
string.IsNullOrEmpty(txtEmail.Text) || string.IsNullOrEmpty(txtPhoneNumber.Text) ||
cboDepartment.SelectedItem == null)
   {
      MessageBox.Show("Please fill in all fields.");
      return;
   }
```

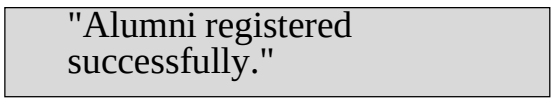
```
    try
    {
        using (SqlConnection con = new SqlConnection(connectionString))
        {
            con.Open();
            string query = "INSERT INTO Alumni (AlumniName, Email, PhoneNumber,
Department) VALUES (@AlumniName, @Email, @PhoneNumber, @Department)";
            using (SqlCommand cmd = new SqlCommand(query, con))
            {
                cmd.Parameters.AddWithValue("@AlumniName", txtAlumniName.Text);
                cmd.Parameters.AddWithValue("@Email", txtEmail.Text);
                cmd.Parameters.AddWithValue("@PhoneNumber", txtPhoneNumber.Text);
                cmd.Parameters.AddWithValue("@Department",
cboDepartment.SelectedItem.ToString());
                cmd.ExecuteNonQuery();
                MessageBox.Show("Alumni registered successfully.");
            }
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show("Error: " + ex.Message);
    }
}
private void btnDisplay_Click(object sender, EventArgs e)
{
    if (cboDepartment.SelectedItem == null)
    {
```

```
        MessageBox.Show("Please select a department.");
        return;
    }
    try
    {
        using (SqlConnection con = new SqlConnection(connectionString))
        {
            con.Open();
            string query = "SELECT AlumniName, Email, PhoneNumber FROM Alumni
WHERE Department = @Department";
            using (SqlCommand cmd = new SqlCommand(query, con))
            {
                cmd.Parameters.AddWithValue("@Department",
cboDepartment.SelectedItem.ToString());
                SqlDataAdapter adapter = new SqlDataAdapter(cmd);
                DataTable dt = new DataTable();
                adapter.Fill(dt);
                dataGridView1.DataSource = dt;
            }
        }
    }
    catch (Exception ex)
    {
        MessageBox.Show("Error: " + ex.Message);
    }
}
public partial class Form1 : Form
{
```

```
    string connectionString = "Data Source=SERVER_NAME;Initial
Catalog=AlumniDB;Integrated Security=True";
    public Form1()
    {
        InitializeComponent();
    }
    private void btnRegister_Click(object sender, EventArgs e)
    {
    }
    private void btnDisplay_Click(object sender, EventArgs e)
    {
    }
}
```

Output:





Alumini Id	Name	Email	Phone	Dept
1	John Doe	johndoe@example.com	1234567890	CS

BY:

SANDHIYA R

73772214191

III – B.E CSE