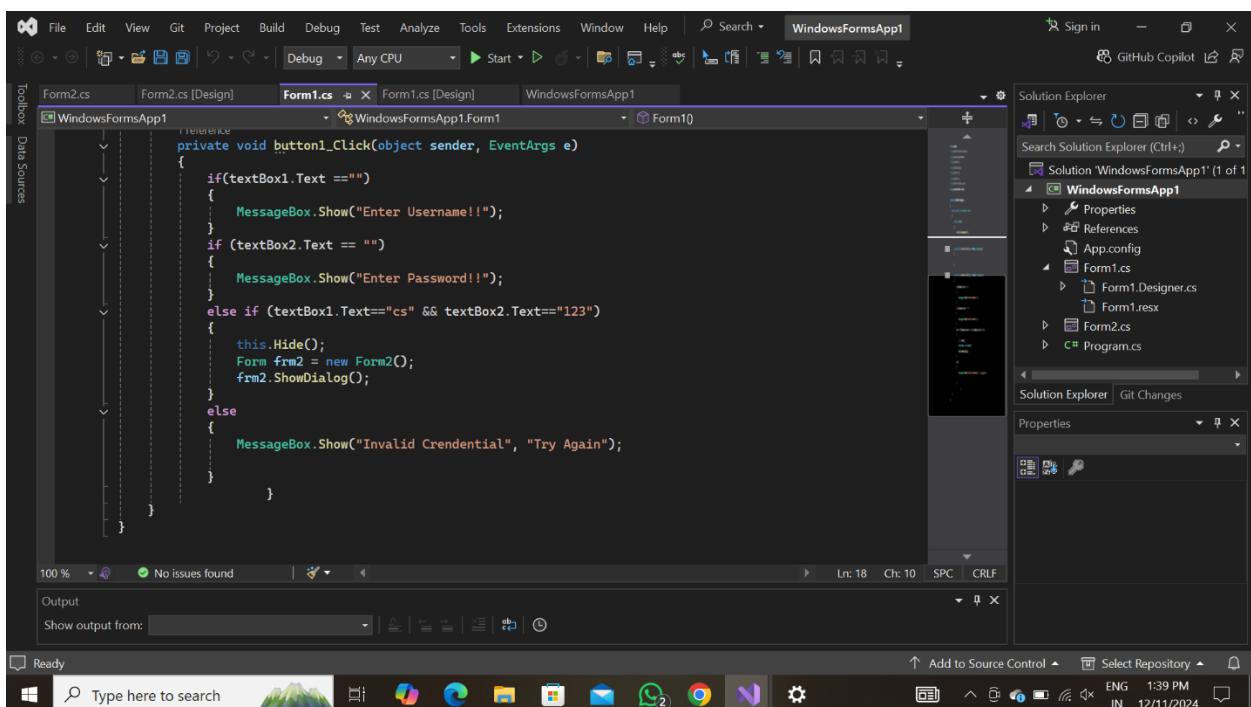
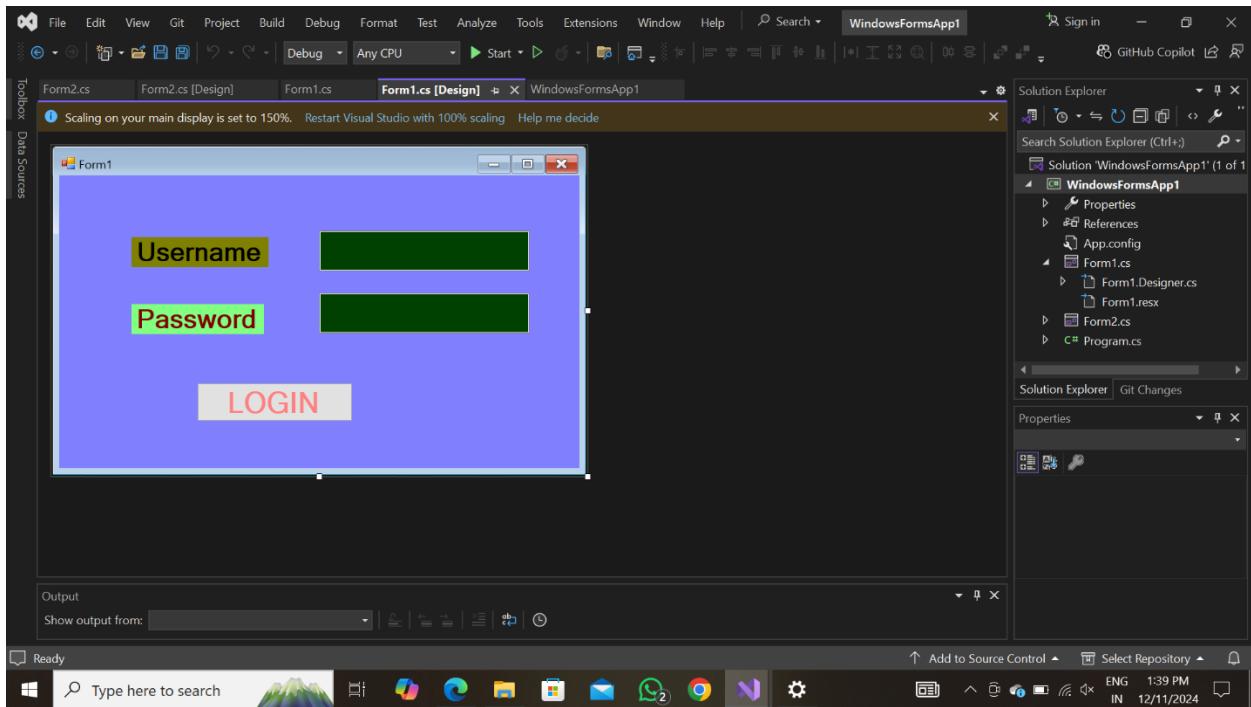
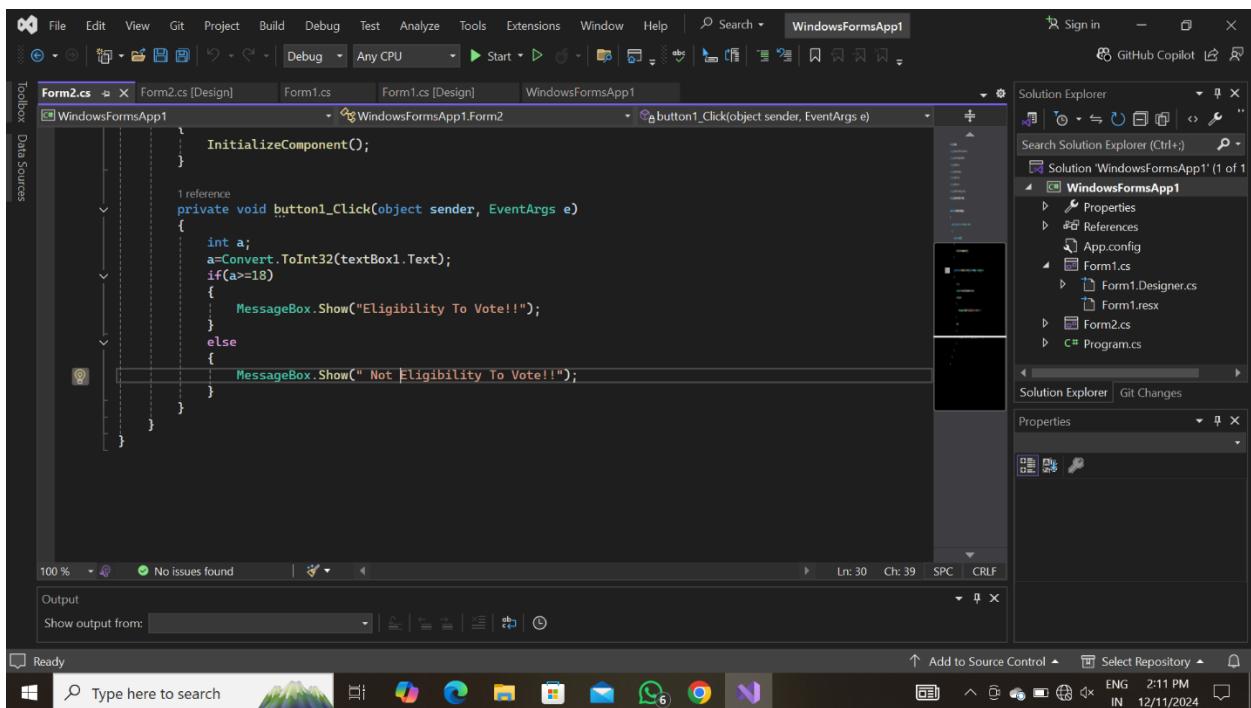
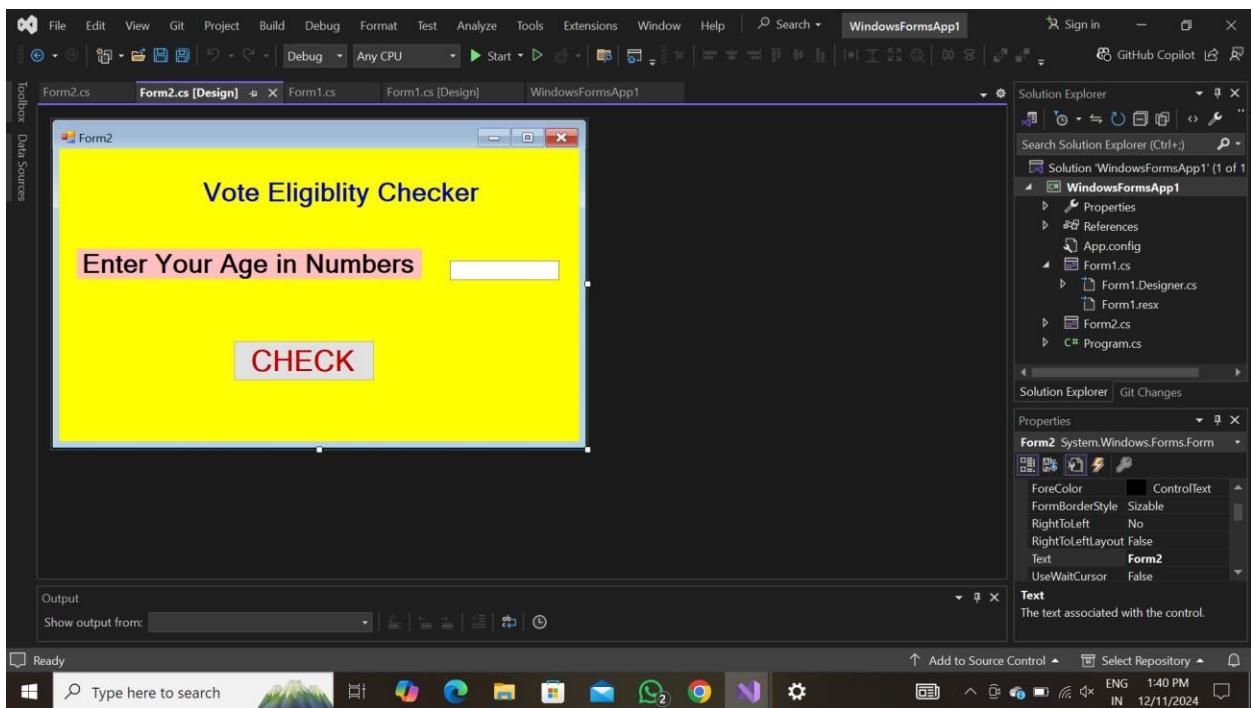
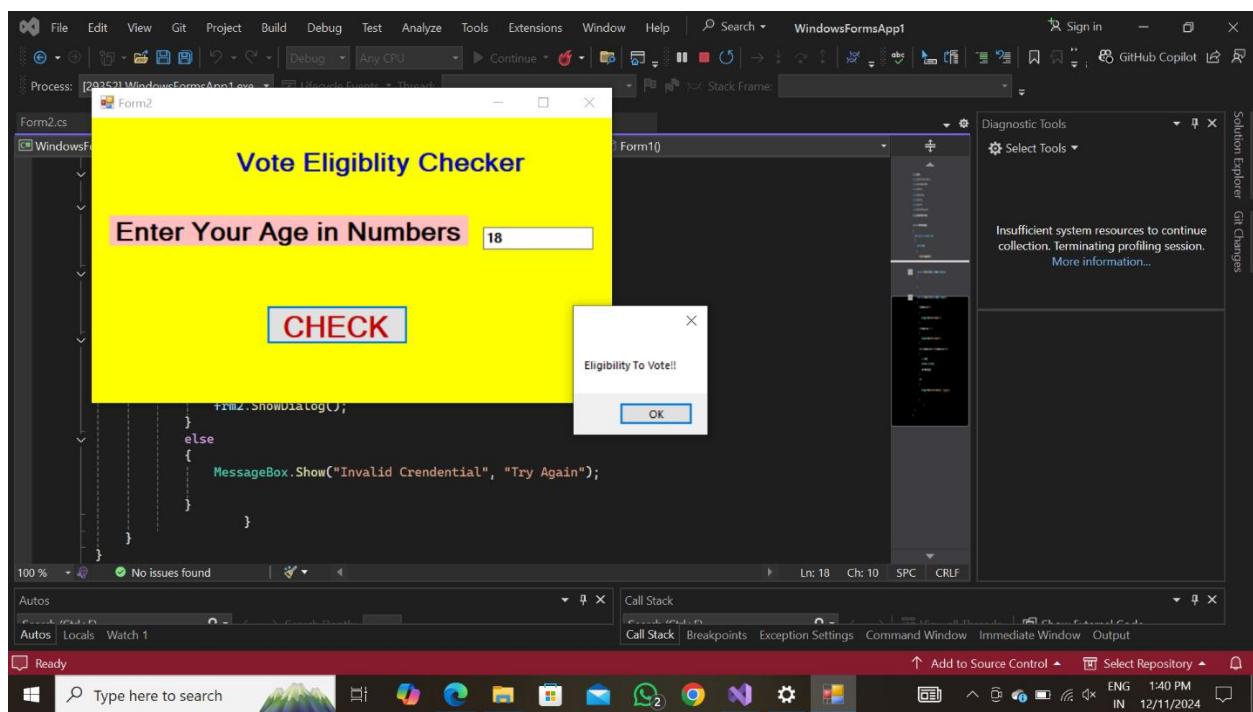
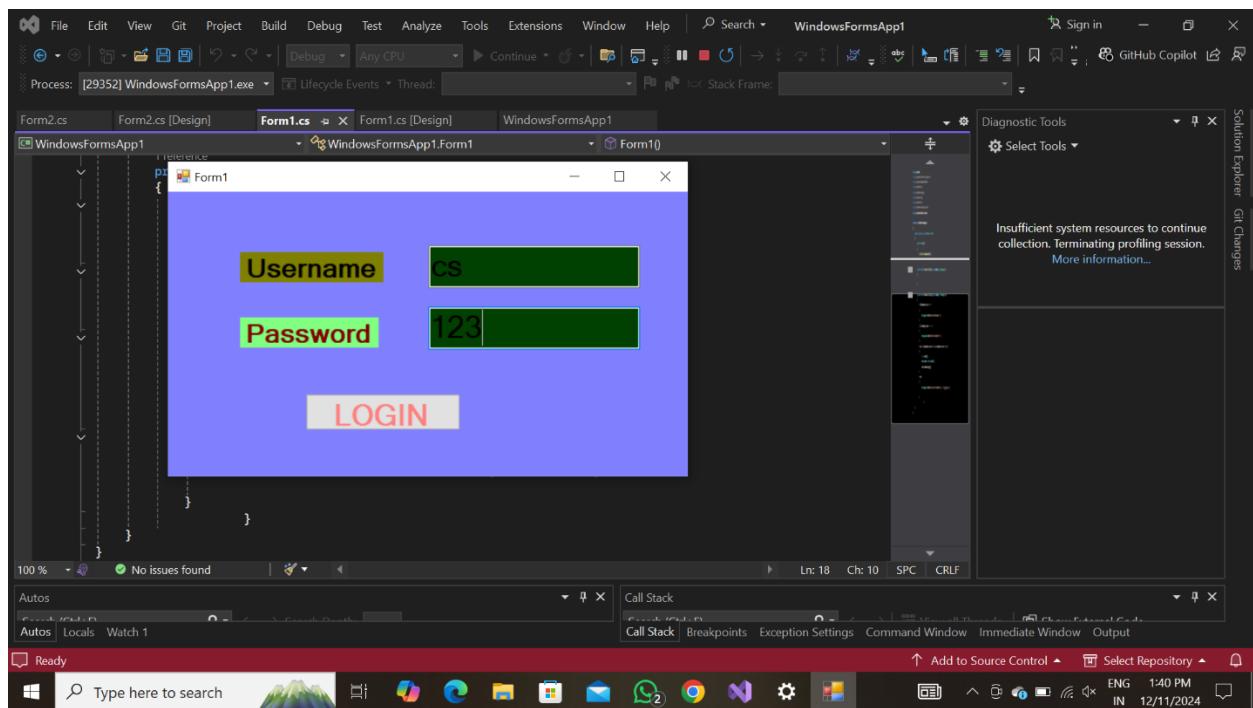
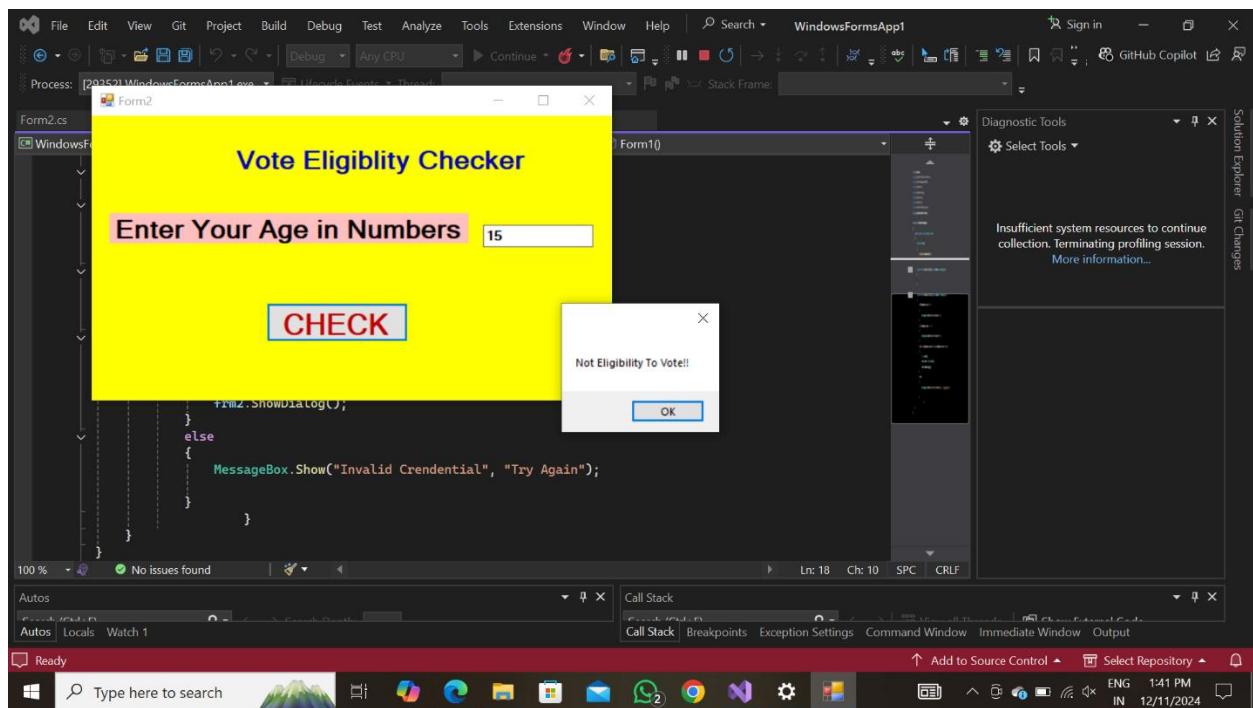


1.EXERCISE USING BASIC SERVER CONTROLS:

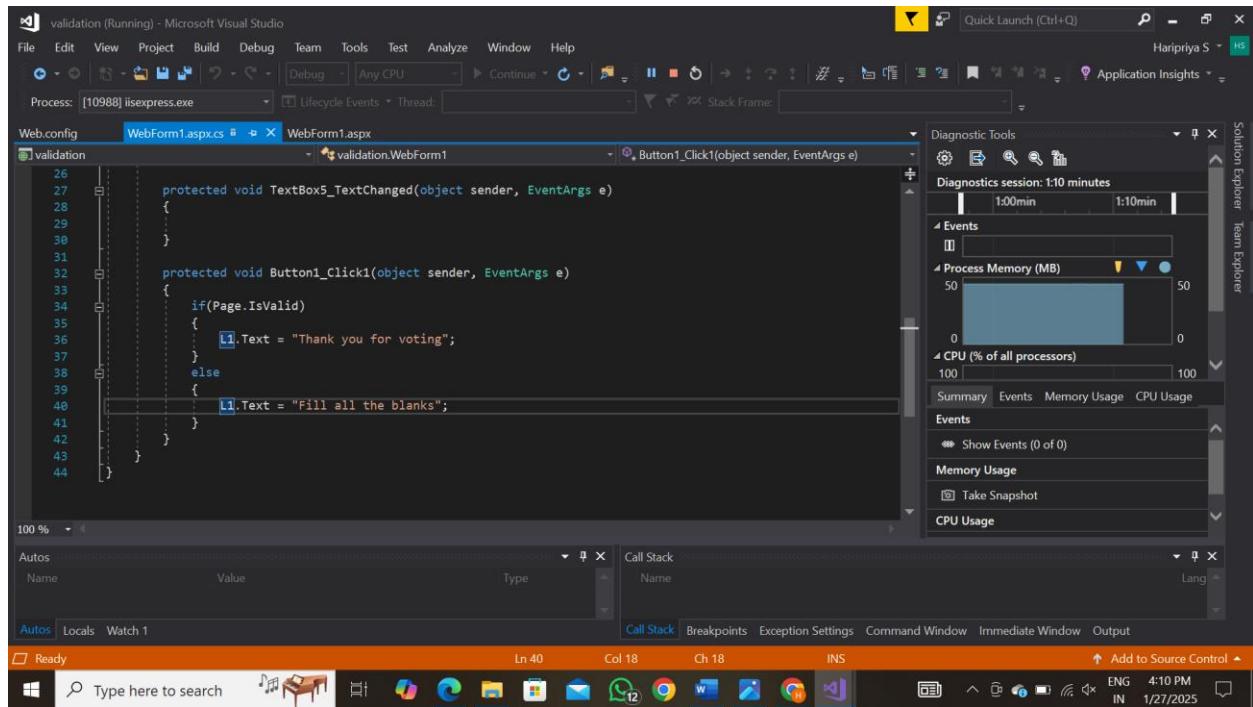








2.EXERCISE USING VALIDATION CONTROLS:



localhost:53064/WebForm1.aspx

COMPUTER SCIENCE DEPARTMENT ELECTION

REGISTER NUMBER : 22BCS040 Thank you for voting

PASSWORD :

CONFIRM PASSWORD : PSGRKCW@123 **PASSWORD AND CONFIRM PASSWORD SHOULD BE SAME**

EMAIL ID : 22bcs040@psgrkcw.ac.in

SECRETARY : John

JOINT SECRETARY : Priya

YEAR OF STUDYING :

I YEAR
 II YEAR
 III YEAR

SUBMIT

Type here to search ENG 4:10 PM IN 1/27/2025

localhost:53064/WebForm1.aspx

COMPUTER SCIENCE DEPARTMENT ELECTION

REGISTER NUMBER : * Fill all the blanks

PASSWORD : PSGRKCW@123

CONFIRM PASSWORD : PSGRKCW@123

EMAIL ID : 22bcs040@psgrkcw.ac.in

SECRETARY : John

JOINT SECRETARY : Priya

YEAR OF STUDYING :

I YEAR
 II YEAR
 III YEAR

SUBMIT

Type here to search ENG 4:11 PM IN 1/27/2025

COMPUTER SCIENCE DEPARTMENT ELECTION

REGISTER NUMBER :

PASSWORD :

CONFIRM PASSWORD :

EMAIL ID :

SECRETARY :

JOINT SECREATRY :

YEAR OF STUDYING :

I YEAR
 II YEAR
 III YEAR

Thank you for voting

3. STUDENT MARK CALCULATION:

```

using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace WindowsFormsApp3
{
    public partial class Form1 : Form
    {
        string[] StudentNames = new string[100];
        int[,] StudentMarks = new int[100, 3];
        int studentCount = 0;
        public Form1()
        {
            InitializeComponent();
        }

        private void panel2_Paint(object sender, PaintEventArgs e)
        {
        }

        private void button1_Click(object sender, EventArgs e)
        {
            if (studentCount >= 100)
            {
                MessageBox.Show("student list full");
            }
            else
            {
                StudentNames[studentCount] = textBox1.Text;
                StudentMarks[studentCount, 0] = Convert.ToInt32(textBox2.Text);
                StudentMarks[studentCount, 1] = Convert.ToInt32(textBox3.Text);
                StudentMarks[studentCount, 2] = Convert.ToInt32(textBox4.Text);
                studentCount++;
                MessageBox.Show("student added successfully");
                ClearInputs();
            }
        }

        private void button2_Click(object sender, EventArgs e)
        {
            Application.Exit();
        }

        private void button3_Click(object sender, EventArgs e)
        {
            Form2 f2 = new Form2();
            f2.Show();
        }
    }
}

```

WindowsFormsApp3 (Running) - Microsoft Visual Studio

File Edit View Project Build Debug Team Test Analyze Window Help

Process [22228] WindowsFormsApp3.exe • Lifecycle Events • Threads

Form1.cs [Design] 4

```
private void ClearInputs()
{
    textBox1.Clear();
    textBox2.Clear();
    textBox3.Clear();
    textBox4.Clear();
}

private void button1_Click(object sender, EventArgs e)
{
    string searchname = textBox1.Text;
    bool found = false;

    for (int i = 0; i < studentcount; i++)
    {
        if (StudentNames[i].Equals(searchname,
            StringComparison.OrdinalIgnoreCase))
        {
            listBox1.Show($"student found: { StudentNames[i] } marks: { StudentMarks[i, 0] }, { StudentMarks[i, 1] }, { StudentMarks[i, 2] }");
            found = true;
            break;
        }
    }

    if (!found)
    {
        MessageBox.Show("student not found.");
    }
}

private void button2_Click(object sender, EventArgs e)
{
    listBox1.Items.Clear();
    for (int i = 0; i < studentcount; i++)
    {
        listBox1.Items.Add($"{StudentNames[i]}-Marks: {StudentMarks[i, 0]}, { StudentMarks[i, 1] }, { StudentMarks[i, 2] }");
    }
}

private void button4_Click(object sender, EventArgs e)
{
    if (studentcount == 0)
    {
        MessageBox.Show("No students available");
        return;
    }

    double totalmarks = 0;
    int totalsubjects = studentcount * 3;
    for (int i = 0; i < studentcount; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            totalmarks += StudentMarks[i, j];
        }
    }
    double average = totalmarks / totalsubjects;
    MessageBox.Show($"class Average Marks: {average:F2}");
}

```

Diagnostic Tools

Diagnostics session: 2:56 minutes

Events

Process Memory (MB)

CPU Usage

Summary Events Memory Usage CPU Usage

Events

Memory Usage

CPU Usage

Autos Locals Watch 1

Call Stack Breakpoints Exception Settings Command Window Immediate Window Output

Ln 1 Col 1 Ch 1 INS

Ready 86°F Sunny 5:14 PM 12/23/2024

WindowsFormsApp3 (Running) - Microsoft Visual Studio

File Edit View Project Build Debug Team Test Analyze Window Help

Process [22228] WindowsFormsApp3.exe • Lifecycle Events • Threads

Form1.cs [Design] 4

```
listBox1.Items.Clear();
for (int i = 0; i < studentcount; i++)
{
    listBox1.Items.Add($"{StudentNames[i]}-Marks: {StudentMarks[i, 0]}, { StudentMarks[i, 1] }, { StudentMarks[i, 2] }");
}

private void button4_Click(object sender, EventArgs e)
{
    if (studentcount == 0)
    {
        MessageBox.Show("No students available");
        return;
    }

    double totalmarks = 0;
    int totalsubjects = studentcount * 3;
    for (int i = 0; i < studentcount; i++)
    {
        for (int j = 0; j < 3; j++)
        {
            totalmarks += StudentMarks[i, j];
        }
    }
    double average = totalmarks / totalsubjects;
    MessageBox.Show($"class Average Marks: {average:F2}");
}

```

Diagnostic Tools

Diagnostics session: 3:00 minutes

Events

Process Memory (MB)

CPU Usage

Summary Events Memory Usage CPU Usage

Events

Memory Usage

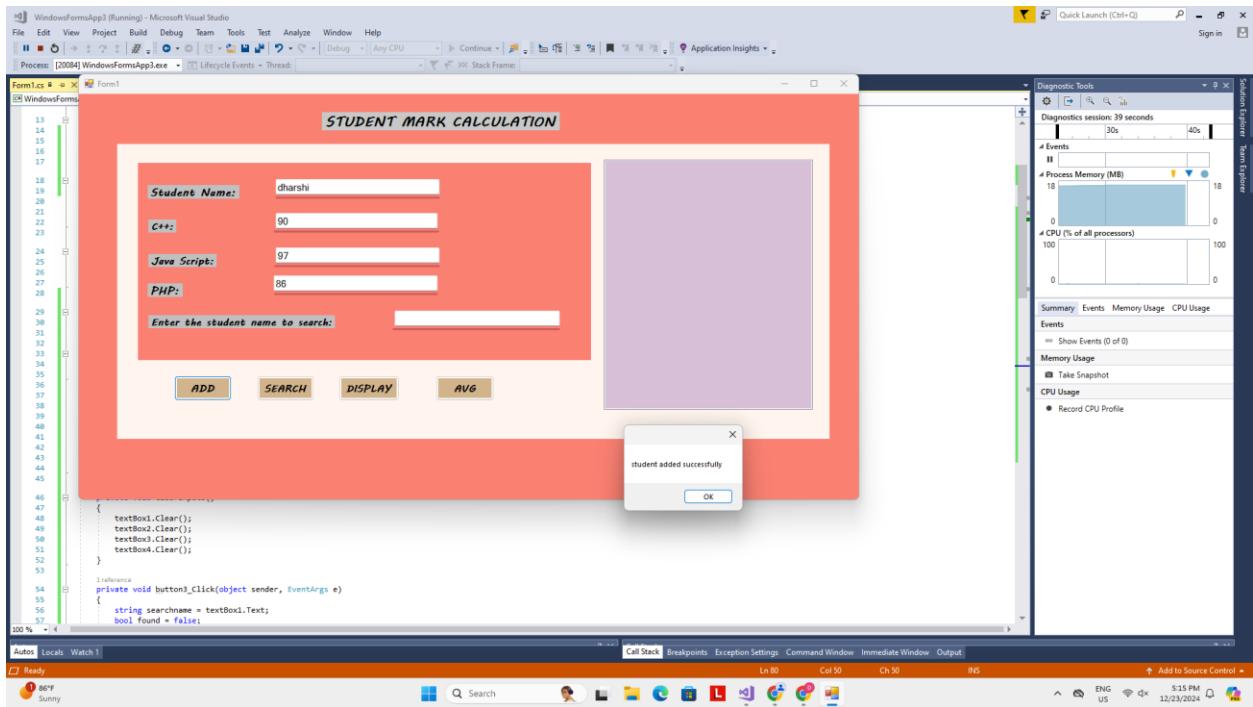
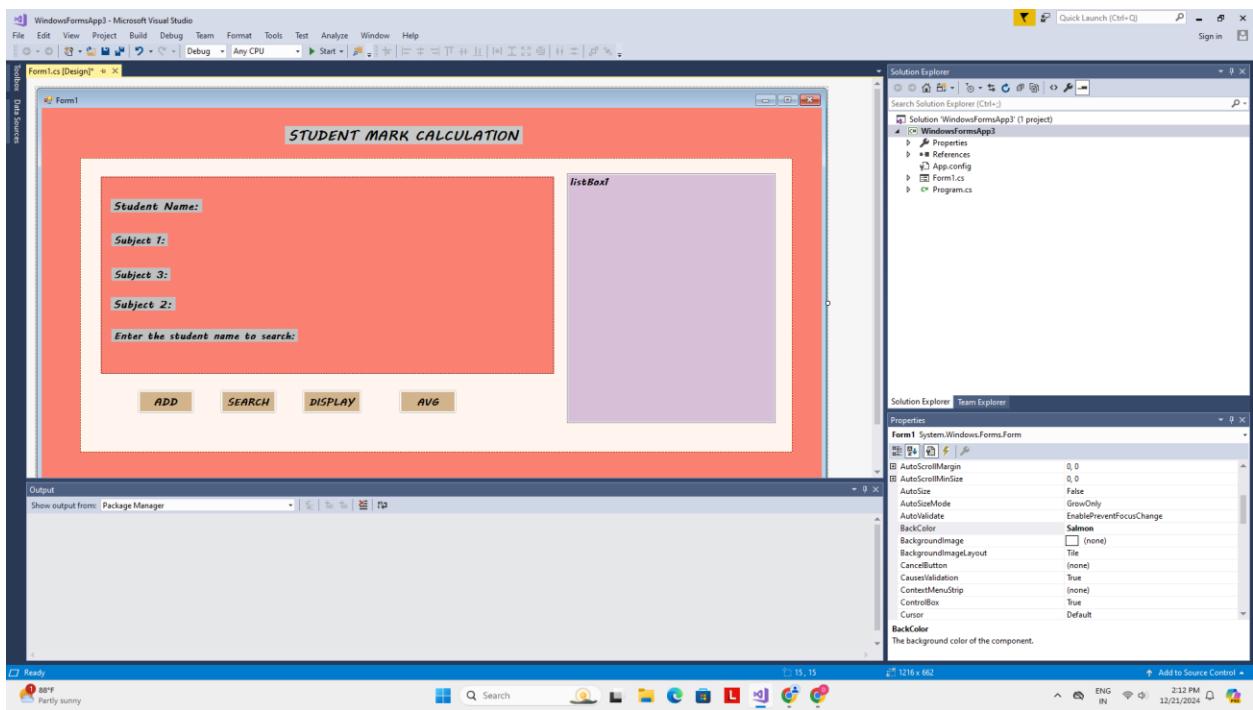
CPU Usage

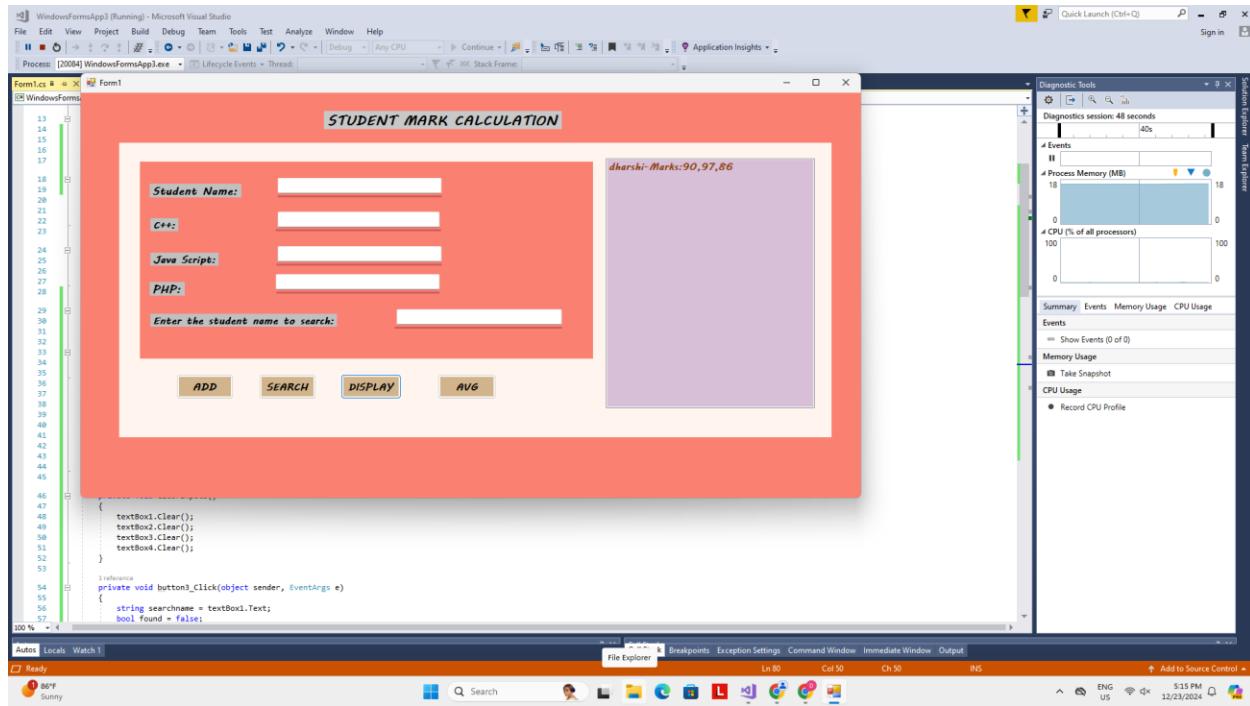
Autos Locals Watch 1

Call Stack Breakpoints Exception Settings Command Window Immediate Window Output

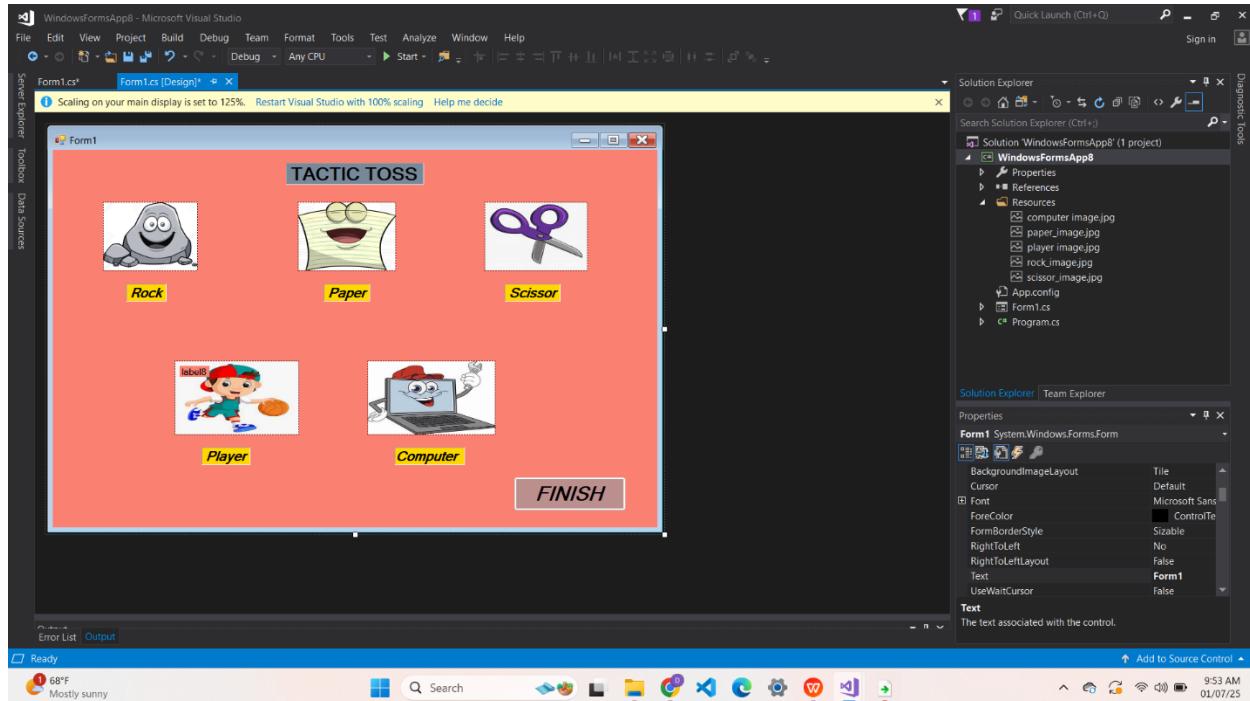
Ln 1 Col 1 Ch 1 INS

Ready 86°F Sunny 5:14 PM 12/23/2024





4.EXERCISE USING ITERATIVE STRUCTURES:



```
WindowsFormsApp8 - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
File -> Debug -> Any CPU -> Start -> Stop -> Refresh -> Find -> Replace -> Properties -> Solution Explorer -> Task List -> Output -> Error List -> Help
Form1.cs [Design]
private void pictureBox1_Click(object sender, EventArgs e)
{
    Random randomGenerator = new Random();
    int computerChoice;
    playerImage = rockImage;
    computerChoice = randomGenerator.Next(1, 4);
    switch (computerChoice)
    {
        case 1:
            computerImage = rockImage;
            Winner.Text = "TIE";
            break;

        case 2:
            computerImage = paperImage;
            Winner.Text = "Computer wins because paper covers rock";
            break;
        case 3:
            computerImage = scissorImage;
            Winner.Text = "Player wins because rock breaks scissors";
            break;
    }
}

private void paperImage_Click(object sender, EventArgs e)
{
    Random randomGenerator = new Random();
    int computerChoice;
    playerImage = paperImage;
    computerChoice = randomGenerator.Next(1, 4);
    switch (computerChoice)
    {
        case 1:
            computerImage = rockImage;
            Winner.Text = " Player wins because paper covers rock ";
            break;
    }
}

private void scissorImage_Click(object sender, EventArgs e)
{
    Random randomGenerator = new Random();
    int computerChoice;
    playerImage = scissorImage;
    computerChoice = randomGenerator.Next(1, 4);
    switch (computerChoice)
    {
        case 1:
            computerImage = rockImage;
            Winner.Text = " Player wins because paper covers rock ";
            break;
        case 2:
            computerImage = paperImage;
            Winner.Text = " Computer wins because scissors cut paper ";
            break;
        case 3:
            computerImage = scissorImage;
            Winner.Text = " TIE ";
            break;
    }
}
```

Solution Explorer

- WindowsFormsApp8 (1 project)
 - Properties
 - References
 - Resources
 - computer image.jpg
 - paper.image.jpg
 - player.image.jpg
 - rock.image.jpg
 - scisor.image.jpg
- App.config
- Form1.cs
- Program.cs

Properties

Diagnostic tools

Search Solution Explorer (Ctrl+F)

100 %

Error List Output

Ready 68°F Mostly sunny

Search

Ln 31 Col 42 Ch 42 INS

10:10 AM 01/07/25

```
WindowsFormsApp8 - Microsoft Visual Studio
File Edit View Project Build Debug Team Tools Test Analyze Window Help
File -> Debug -> Any CPU -> Start -> Stop -> Refresh -> Find -> Replace -> Properties -> Solution Explorer -> Task List -> Output -> Error List -> Help
Form1.cs [Design]
private void pictureBox1_Click(object sender, EventArgs e)
{
    Random randomGenerator = new Random();
    int computerChoice;
    playerImage = rockImage;
    computerChoice = randomGenerator.Next(1, 4);
    switch (computerChoice)
    {
        case 1:
            computerImage = rockImage;
            Winner.Text = " Player wins because paper covers rock ";
            break;
        case 2:
            computerImage = paperImage;
            Winner.Text = " Computer wins because paper covers rock ";
            break;
        case 3:
            computerImage = scissorImage;
            Winner.Text = " Player wins because scissors cut paper ";
            break;
    }
}

private void paperImage_Click(object sender, EventArgs e)
{
    Random randomGenerator = new Random();
    int computerChoice;
    playerImage = paperImage;
    computerChoice = randomGenerator.Next(1, 4);
    switch (computerChoice)
    {
        case 1:
            computerImage = rockImage;
            Winner.Text = " Computer wins because rock breaks scissors ";
            break;
        case 2:
            computerImage = paperImage;
            Winner.Text = " Player wins because scissors cut paper ";
            break;
        case 3:
            computerImage = scissorImage;
            Winner.Text = " TIE ";
            break;
    }
}

private void scissorImage_Click(object sender, EventArgs e)
{
    Random randomGenerator = new Random();
    int computerChoice;
    playerImage = scissorImage;
    computerChoice = randomGenerator.Next(1, 4);
    switch (computerChoice)
    {
        case 1:
            computerImage = rockImage;
            Winner.Text = " Computer wins because rock breaks scissors ";
            break;
        case 2:
            computerImage = paperImage;
            Winner.Text = " Player wins because scissors cut paper ";
            break;
        case 3:
            computerImage = scissorImage;
            Winner.Text = " TIE ";
            break;
    }
}
```

Solution Explorer

- WindowsFormsApp8 (1 project)
 - Properties
 - References
 - Resources
 - computer image.jpg
 - paper.image.jpg
 - player.image.jpg
 - rock.image.jpg
 - scisor.image.jpg
- App.config
- Form1.cs
- Program.cs

Properties

Diagnostic tools

Search Solution Explorer (Ctrl+F)

100 %

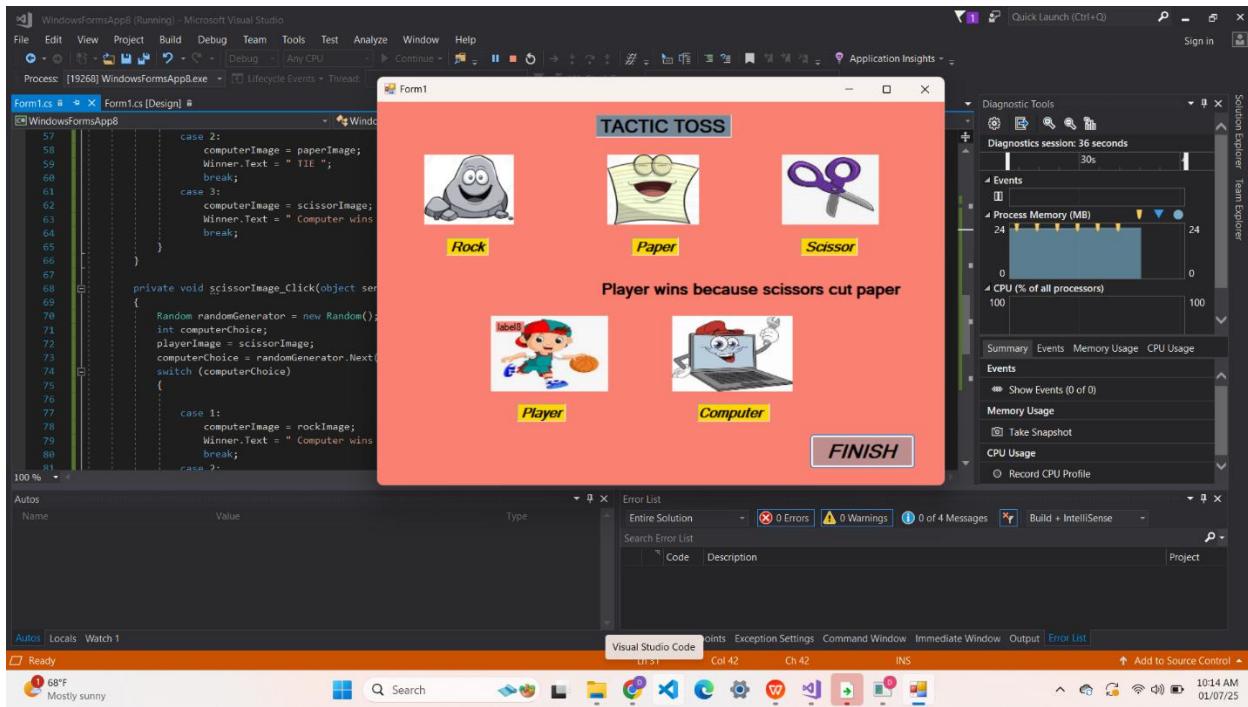
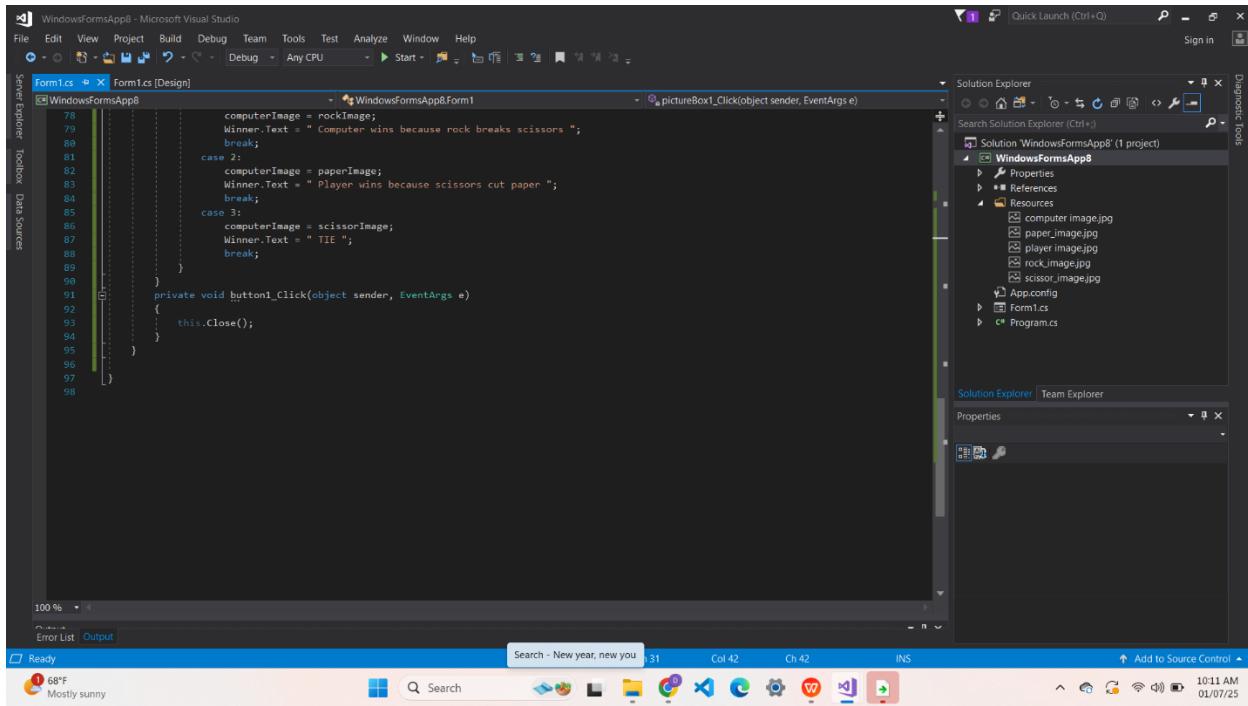
Error List Output

Ready 68°F Mostly sunny

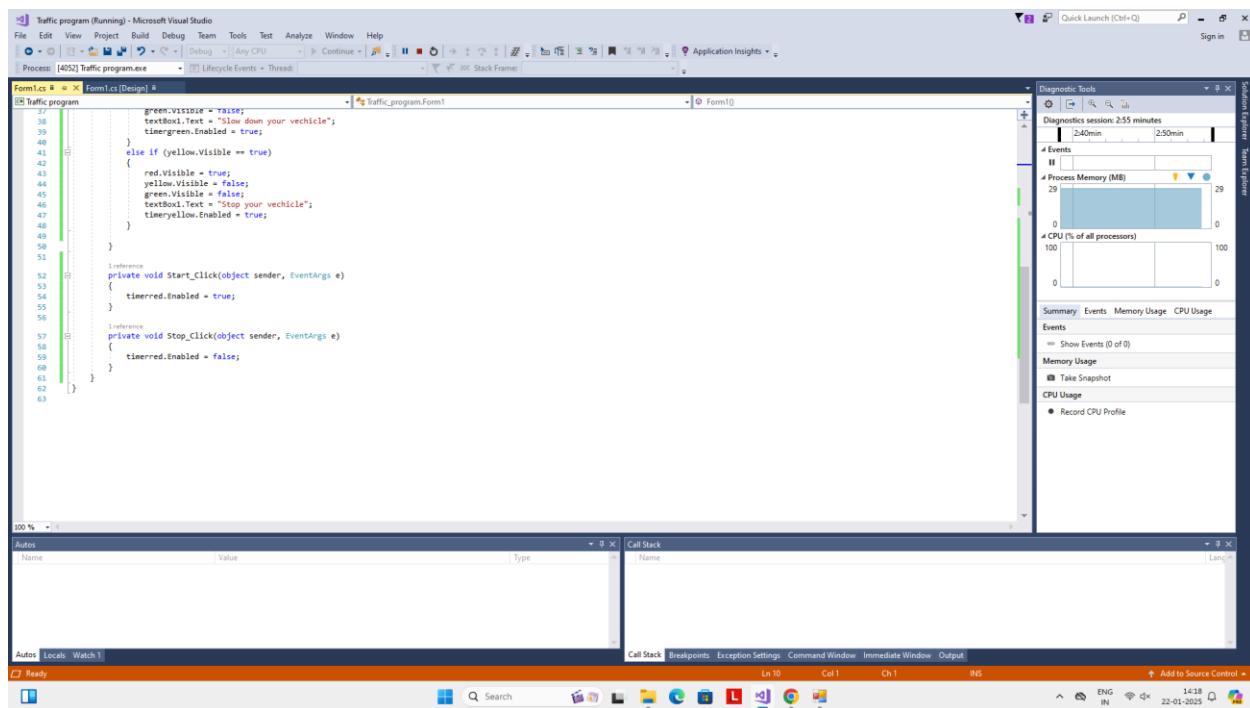
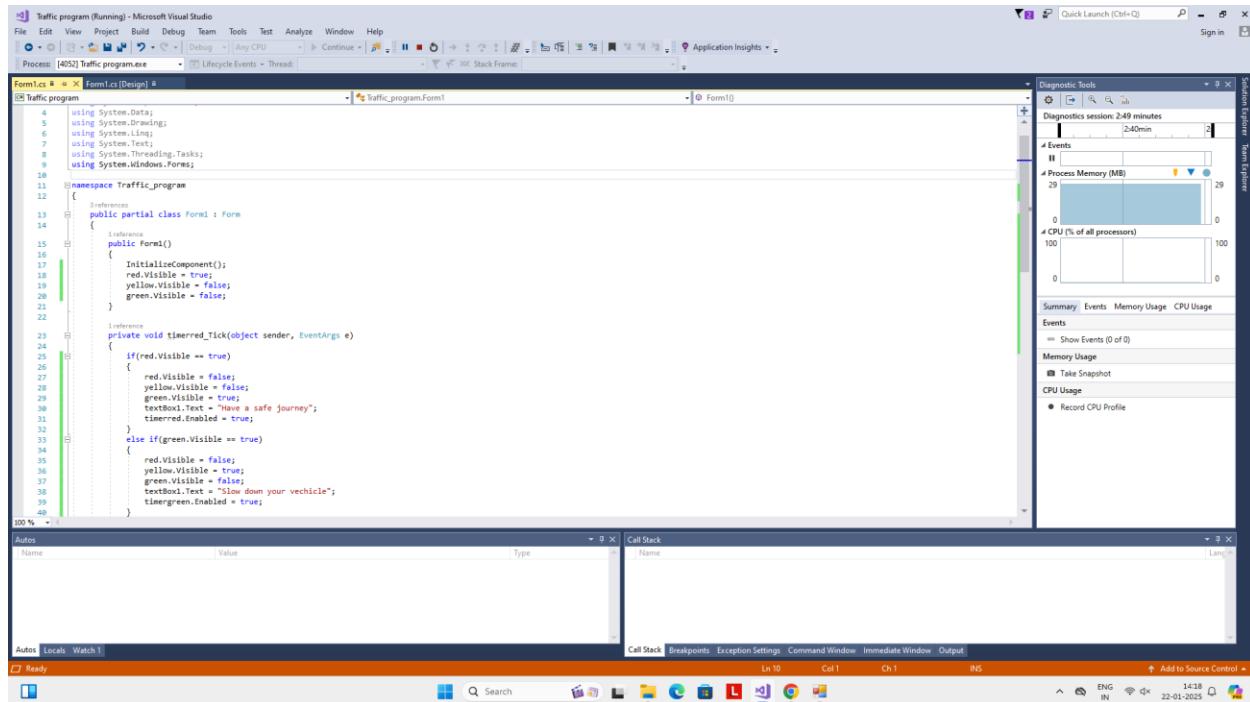
Search

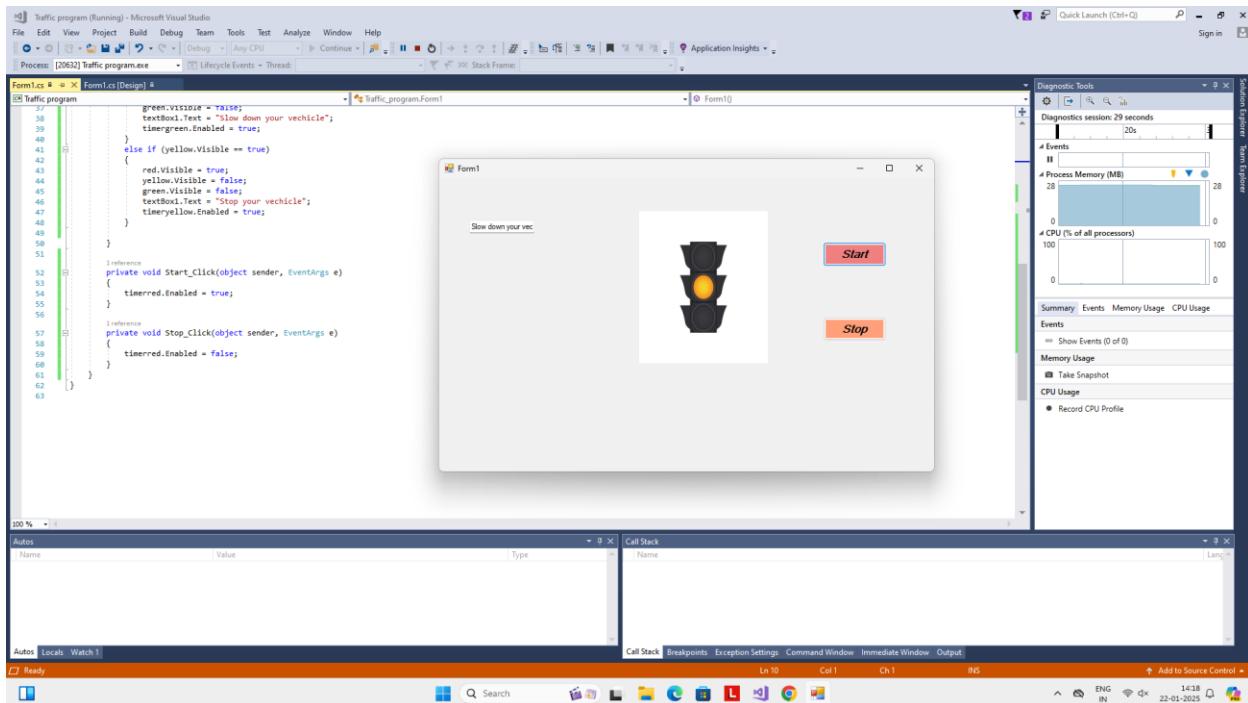
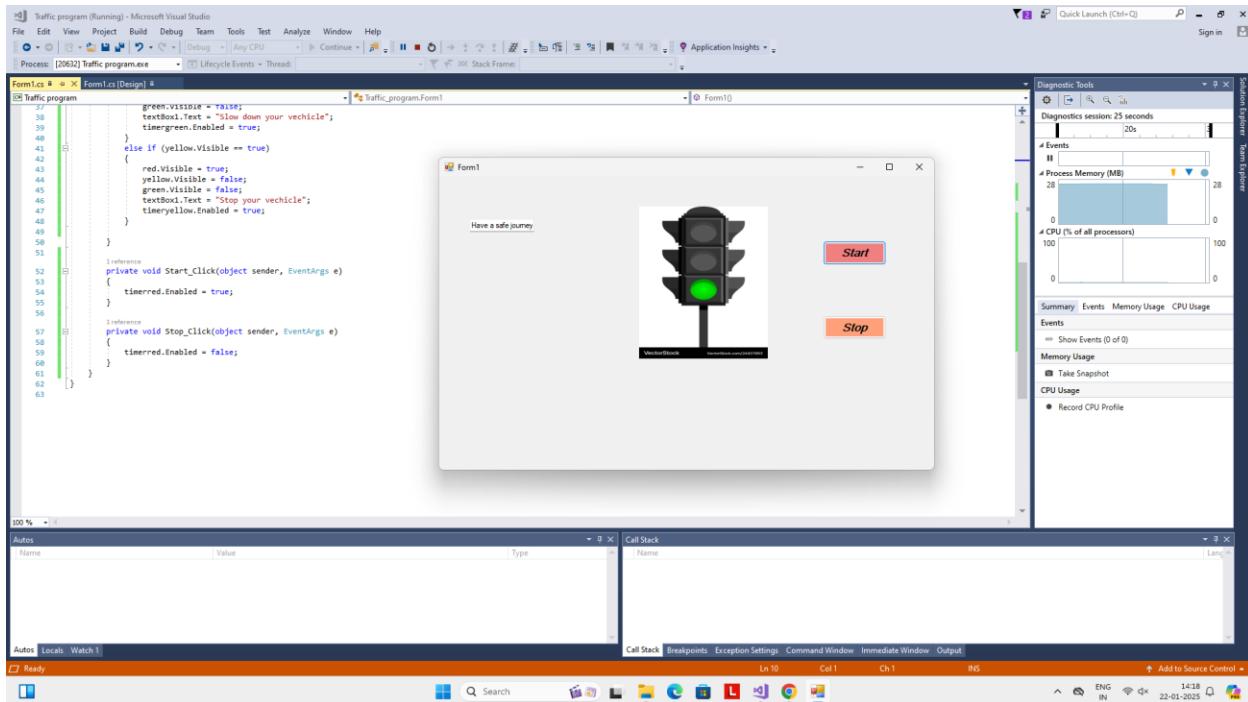
Ln 31 Col 42 Ch 42 INS

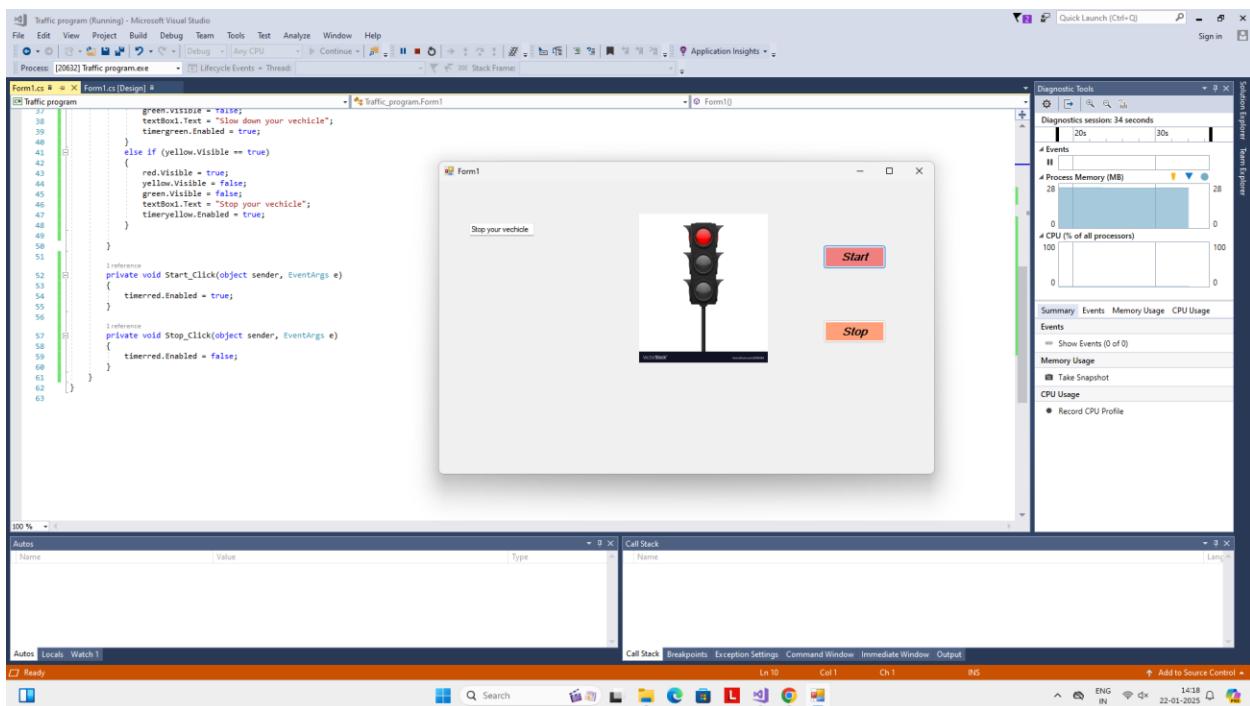
10:10 AM 01/07/25



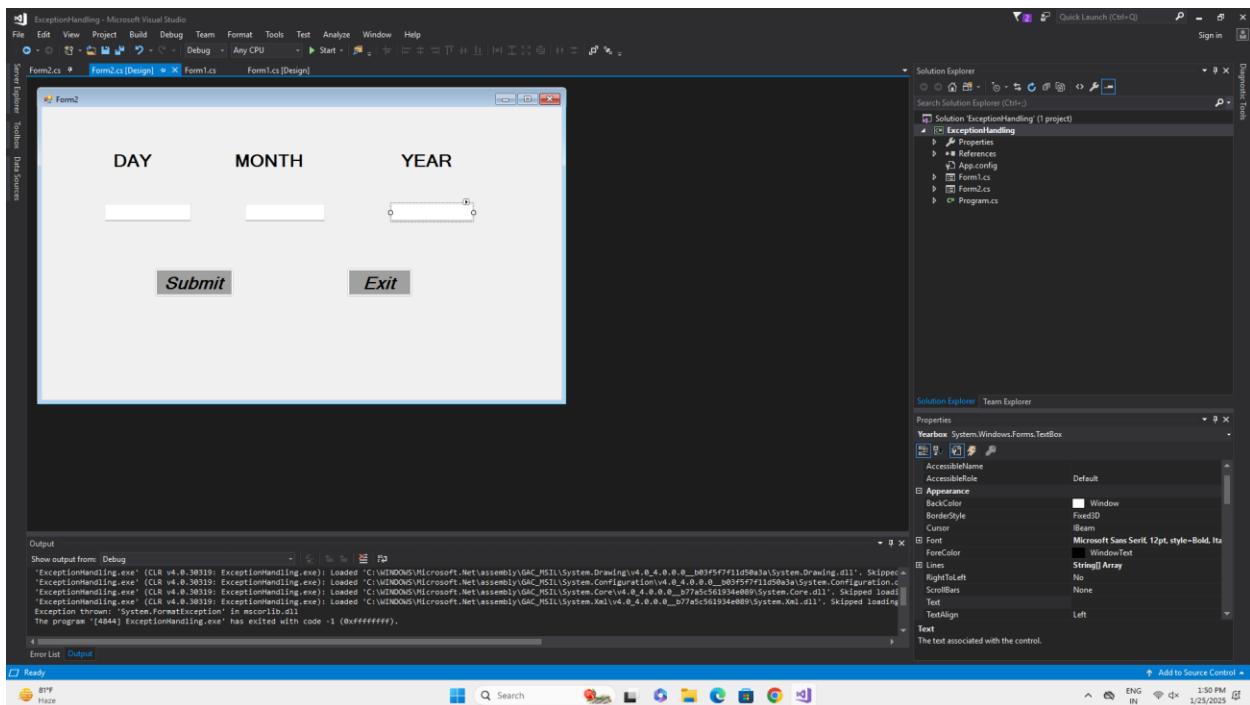
6.EXERCISE USING BASIC CONTROL CLASSES:







7.EXERCISE USING EXCEPTION HANDLING:



ExceptionHandling - Microsoft Visual Studio

File Edit View Project Build Debug Tools Test Analyze Window Help

Form2.cs [Design] Form1.cs [Design] ExceptionHandling.cs

Solution Explorer

Server Explorer

Tickets

Data Sources

```

private void button2_Click(object sender, EventArgs e)
{
    Form2 form2 = new Form2();
    form2.Show();
    this.Hide();
}

private void textBox1_TextChanged(object sender, EventArgs e)
{
}

private void button1_Click(object sender, EventArgs e)
{
    try
    {
        int numerator = Convert.ToInt32(textBox1.Text);
        int denominator = Convert.ToInt32(textBox2.Text);
        int answer = numerator / denominator;
        textBox3.Text = answer.ToString();
    }
    catch (FormatException)
    {
        MessageBox.Show("Invalid input enter two numbers please");
    }
    catch (DivideByZeroException divideByZeroException)
    {
        MessageBox.Show(divideByZeroException.Message, "Can't divide by zero");
    }
    finally
    {
        textBox1.Text = "";
        textBox2.Text = "";
    }
}

```

Output

```

Show output from: Debug
ExceptionHandling.exe (CLR v4.0.30319: ExceptionHandling.exe): Loaded 'C:\Windows\Microsoft.NET\assembly\GAC_MSIL\System.Drawing\v4.0.0.0__b03f5f7f11d50a3a\System.Drawing.dll'. Skipped loading because it is located in the GAC.
ExceptionHandling.exe (CLR v4.0.30319: ExceptionHandling.exe): Loaded 'C:\Windows\Microsoft.NET\assembly\GAC_MSIL\System.Configuration\v4.0.0.0__b03f5f7f11d50a3a\System.Configuration.dll'.
ExceptionHandling.exe (CLR v4.0.30319: ExceptionHandling.exe): Loaded 'C:\Windows\Microsoft.NET\assembly\GAC_MSIL\System.Core\v4.0.0.0__b77a5c561934e089\System.Core.dll'. Skipped loading because it is located in the GAC.
ExceptionHandling.exe (CLR v4.0.30319: ExceptionHandling.exe): Loaded 'C:\Windows\Microsoft.NET\assembly\GAC_MSIL\System.Xml\v4.0.0.0__b77a5c561934e089\System.Xml.dll'. Skipped loading because it is located in the GAC.
Exception thrown: 'System.FormatException' in mscorelib.dll
The program '[4044] ExceptionHandling.exe' has exited with code -1 (0xffffffff).

```

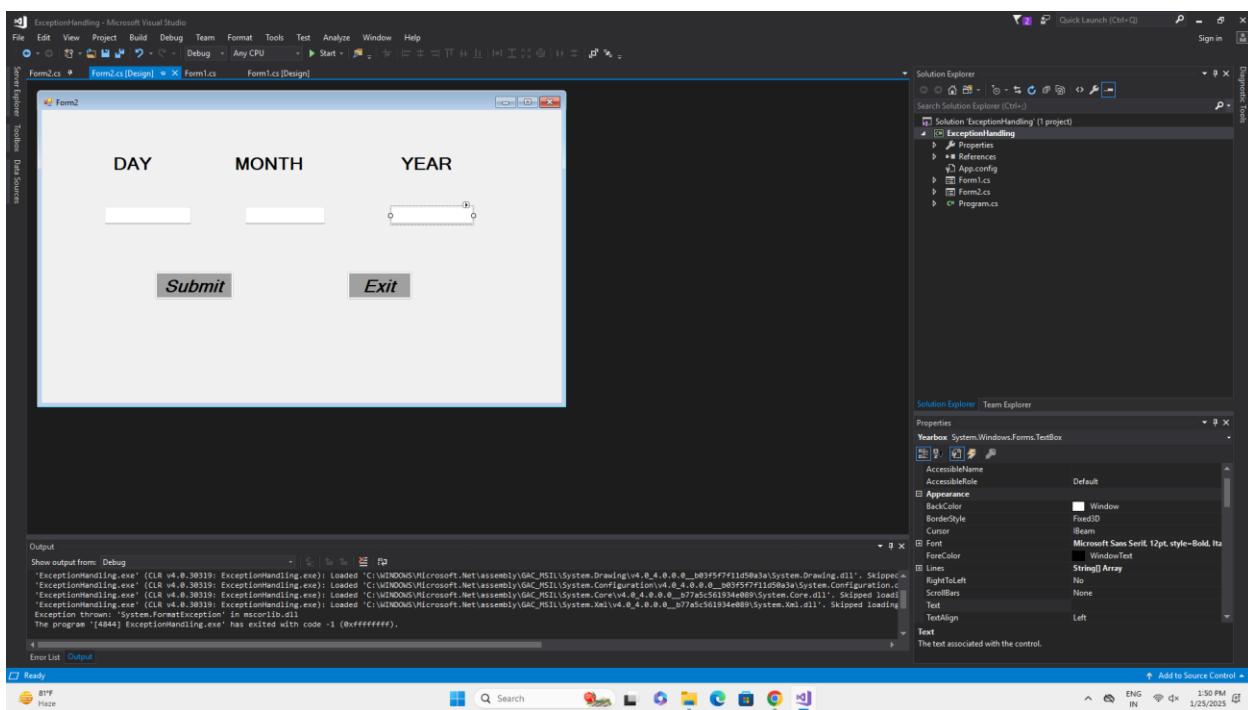
Error List Output

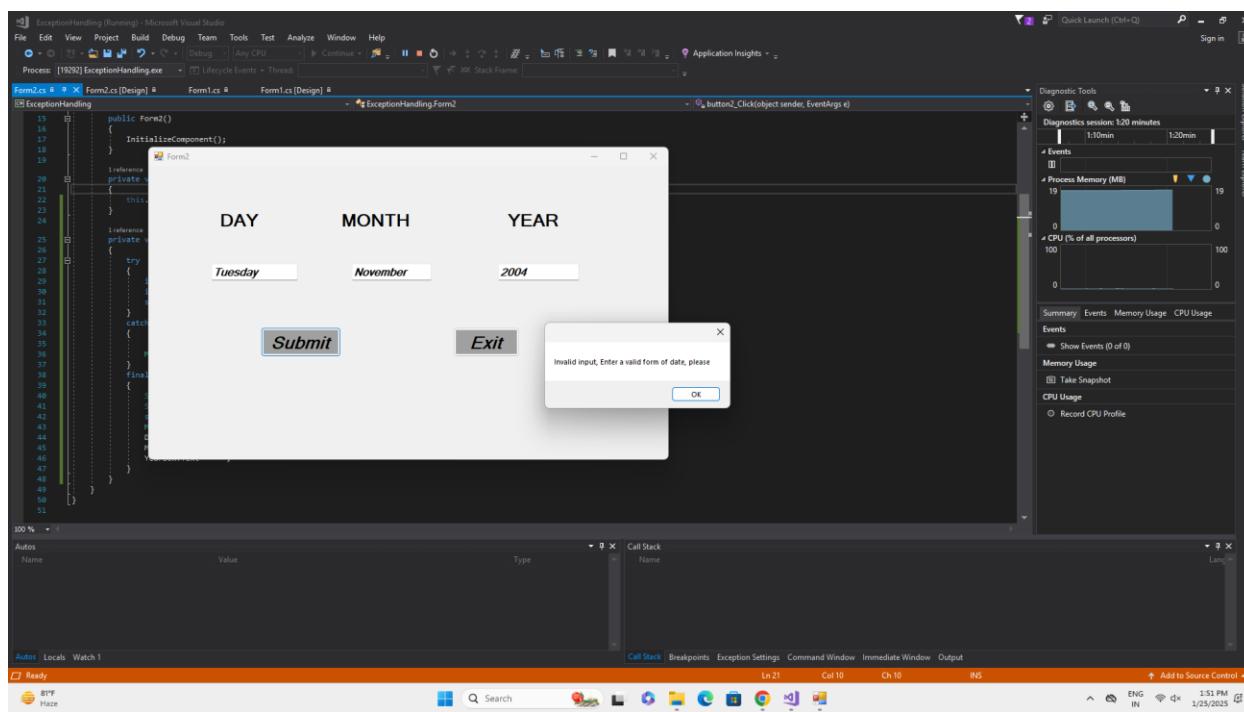
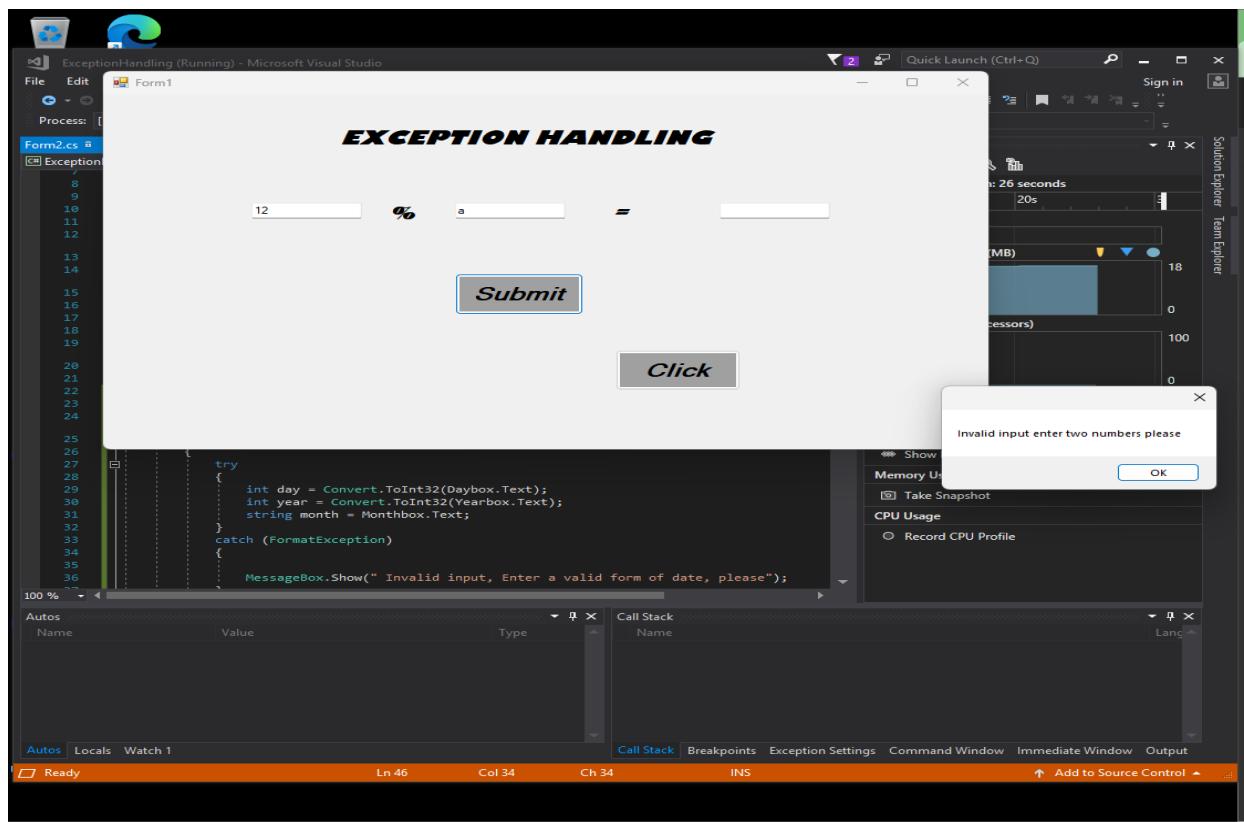
Ready

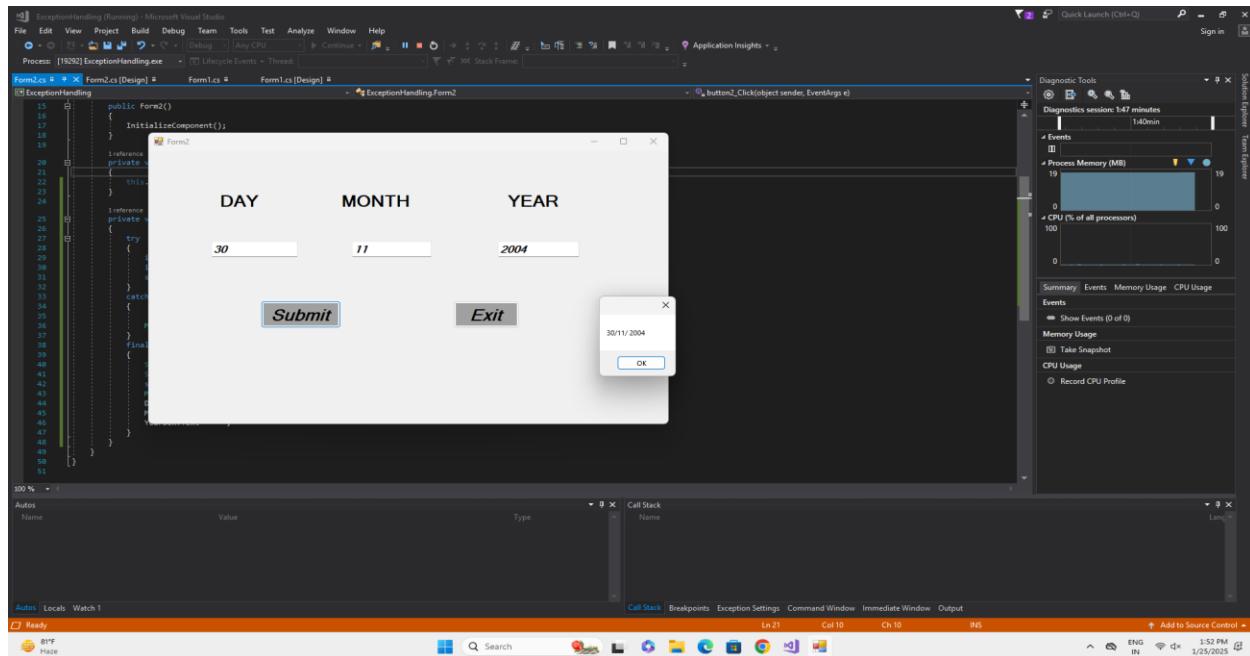
BTW Haze

Search

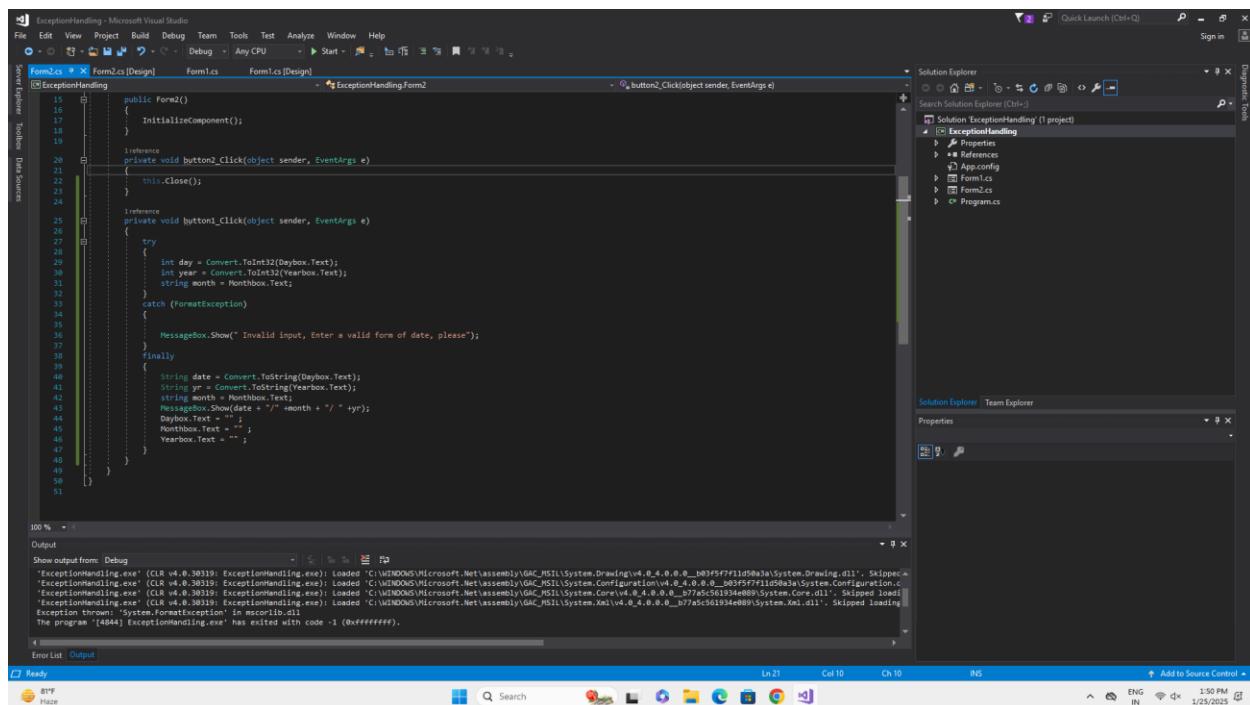
1 ENG IN 1:48 PM 1/25/2025







8.STRING MANIPULATION:



A screenshot of Microsoft Visual Studio showing a C# application named "String". The code in the editor handles user input for a string and performs various manipulations based on user choice. The application outputs the original string, its length, and counts of capital letters, small letters, numbers, and total spaces. It also provides a menu for string manipulation functions like ToUpper, ToLower, Trim, Replace, etc. The output window shows the results of the manipulation.

```
int cn = int.Parse(choice);
string output = "";
switch (ch)
{
    case 1:
        string b = (string)s.Clone();
        output = b;
        Console.WriteLine(" Output: " + output);
        break;
    case 2:
        Console.WriteLine(" Enter target substring: ");
        string tar = Console.ReadLine();
        Console.WriteLine(s.Contains(tar));
        break;
    case 3:
        output = s.ToLower();
        Console.WriteLine(" Output: " + output);
        break;
    case 4:
        output = s.ToUpper();
        Console.WriteLine(" Output: " + output);
        break;
    case 5:
        output = s.Trim();
        Console.WriteLine(" Output: " + output);
        break;
    case 6:
        Console.WriteLine(" Enter start index: ");
        int start = int.Parse(Console.ReadLine());
        Console.WriteLine(" Enter length: ");
        int length = int.Parse(Console.ReadLine());
        output = s.Substring(start, length);
        Console.WriteLine(" Output: " + output);
        break;
    case 7:
        Console.WriteLine(" Enter target substring: ");
        string target = Console.ReadLine();
        Console.WriteLine("Enter replacement substring: ");
        string replacement = Console.ReadLine();
        output = s.Replace(target, replacement);
        Console.WriteLine(" Output: " + output);
        break;
    default:
        Console.WriteLine(" Invalid choice.");
        break;
}
Console.WriteLine(" Press any key to exit.");
Console.ReadKey();
```

A second screenshot of Microsoft Visual Studio showing the same C# application. The output window shows a different string ("Hello world @123 %%") and the same analysis of capital letters, small letters, numbers, and spaces. The manipulation menu is identical.

```
int cn = int.Parse(choice);
string output = "";
switch (ch)
{
    case 1:
        string b = (string)s.Clone();
        output = b;
        Console.WriteLine(" Output: " + output);
        break;
    case 2:
        output = s.ToUpper();
        Console.WriteLine(" Output: " + output);
        break;
    case 3:
        output = s.ToLower();
        Console.WriteLine(" Output: " + output);
        break;
    case 4:
        output = s.Trim();
        Console.WriteLine(" Output: " + output);
        break;
    case 5:
        Console.WriteLine(" Enter start index: ");
        int start = int.Parse(Console.ReadLine());
        Console.WriteLine(" Enter length: ");
        int length = int.Parse(Console.ReadLine());
        output = s.Substring(start, length);
        Console.WriteLine(" Output: " + output);
        break;
    case 6:
        Console.WriteLine(" Enter target substring: ");
        string target = Console.ReadLine();
        Console.WriteLine("Enter replacement substring: ");
        string replacement = Console.ReadLine();
        output = s.Replace(target, replacement);
        Console.WriteLine(" Output: " + output);
        break;
    default:
        Console.WriteLine(" Invalid choice.");
        break;
}
Console.WriteLine(" Press any key to exit.");
Console.ReadKey();
```

