

Practical No. 1

Basic Of C#

Q.1 Design a window application to demonstrate basic and advanced controls. Create Registration form with following fields first name, middle name, surname, photo, dob, address, mobile no, username, password, gender, hobbies, city. Use appropriate controls to take inputs. And show the entered data on another form.

User Interface:

The screenshot shows a Windows application window titled "Form1". The main heading is "Registration Form". The form contains the following fields and controls:

- First Name: Text box with "Bhushan"
- Middle Name: Text box with "D."
- Last Name: Text box with "Shirsat"
- User Name: Text box with "BhusanDDS"
- Password: Text box with "113"
- DOB: Date picker showing "Wednesday, February"
- City: Text box with "Goa"
- Mobile No.: Text box with "98706543"
- Address: Text box with "Goa"
- Hobbies: Text box with "Book Reading"
- Photo: A small image of a person, with an "Upload" button below it.
- Gender: Radio buttons for "Male" (selected) and "Female"
- Buttons: "Register" and "Close" at the bottom.

Program:

Form1.cs

```
namespace PractNo1Question1
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            this.Close();
        }

        private void button2_Click(object sender, EventArgs e)
        {
        }
```

```

Form2 f2 = new Form2();

f2.fname = txtfname.Text;
f2.mname = txtmname.Text;
f2.lname = txtLname.Text;
f2.dob = dtDOB.Value.ToShortDateString();
f2.add = txtAdd.Text;
f2.mob = txtMob.Text;
f2.uname = uname.Text;
f2.pass = password.Text;
f2.gen = radioBtnMale.Checked ? "Male" : "Female";
f2.hobby = txtHobby.Text;
f2.city = txtCity.Text;
f2.imgB = picBox.Image;
f2.ShowDialog();
}

private void UpBtn_Click(object sender, EventArgs e)
{
    String imageLocation = "";
    try
    {
        OpenFileDialog dialog = new OpenFileDialog();
        dialog.Filter = "Image Files (*.jpg; *.jpeg; *.png; *.gif; *.bmp)|*.jpg; *.jpeg; *.png; *.gif; *.bmp";

        if (dialog.ShowDialog() == System.Windows.Forms.DialogResult.OK)
        {
            imageLocation = dialog.FileName;

            picBox.ImageLocation = imageLocation;
        }

    } catch (Exception)
    {
        MessageBox.Show("Error:" ,
            "Error", MessageBoxButtons.OK, MessageBoxIcon.Error);
    }
}
}
}

```

Form2.cs

```

namespace PractNo1Question1
{
    public partial class Form2 : Form
    {
        public string fname { get; set; }
        public string mname { get; set; }
        public string lname { get; set; }
        public string uname { get; set; }
        public string pass { get; set; }
        public string city { get; set; }
        public string add { get; set; }
        public string hobby { get; set; }
        public string mob { get; set; }
    }
}

```

```

public string dob { get; set; }
public Image imgB { get; set; }
public string gen { get; set; }

public Form2()
{
    InitializeComponent();
}
private void Form2_Load(object sender, EventArgs e)
{
    lblFname.Text = fname;
    lblMname.Text = mname;
    lblLname.Text = lname;
    lbluname.Text = unname;
    lblPass.Text = pass;
    lblCity.Text = city;
    lblAdd.Text = add;
    lblGen.Text = gen;
    lblHobby.Text = hobby;
    lblMob.Text = mob;
    lblDOB.Text = dob;
    pictureBox1.Image = imgB;
}
}
}

```

Program.cs

```

static void Main()
{
    Application.EnableVisualStyles();
    Application.SetCompatibleTextRenderingDefault(false);
    Application.Run(new Form1());
}

```

Output:

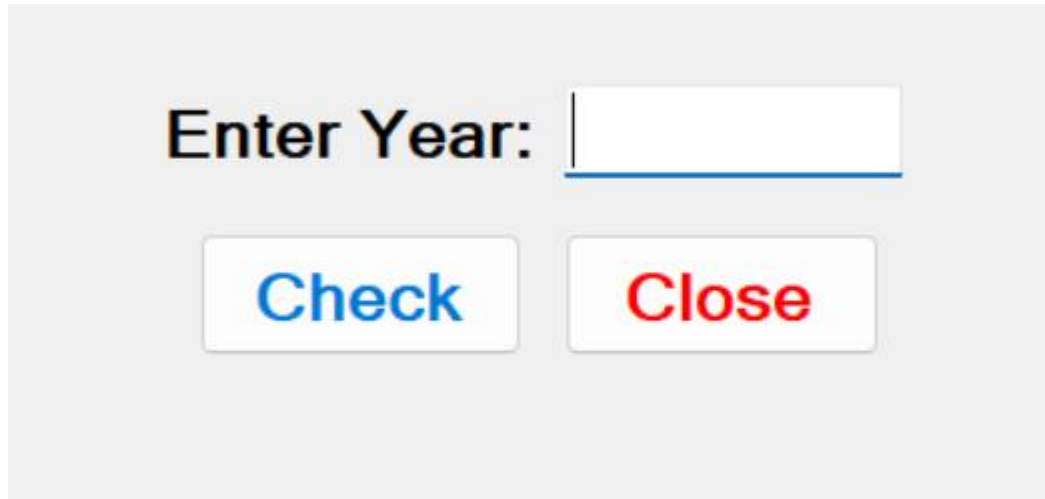


The screenshot shows a Windows application window titled "Form2". Inside the window, there is a form titled "Person Information" in red text. The form displays the following details:

| | | |
|---------------|--------------------|---|
| Name | Bhushan D. Shirsat |  |
| Date Of Birth | 2/7/2024 | |
| Mobile No | 98706543 | |
| Address | Goa | |
| User Name | BhusanDDS | |
| Password | 113 | |
| Gender | Male | |
| City | Goa | |
| Hobbies | Book Reading | |

Q.2 Design a window application in c# using objects and classes to find whether an entered year is leap year or not.

User Interface:

The image shows a simple Windows Forms application window. It has a light gray background. At the top, the text "Enter Year:" is displayed in a large, bold, black font. To the right of this text is a white text box with a blue border. Below the text box, there are two buttons. The first button is labeled "Check" in blue text on a white background. The second button is labeled "Close" in red text on a white background. Both buttons have a slight 3D effect with a gray shadow.

PractNo1Q2.cs

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

```
namespace PracticalNo1All
```

```
{
    public partial class Pract1Q2 : Form
    {
        Year y = null;
        public Pract1Q2()
        {
            InitializeComponent();
        }

        private void Pract1Q2_Load(object sender, EventArgs e)
        {
        }

        private void btnCheck_Click(object sender, EventArgs e)
        {
            try
            {
                int e2 = Convert.ToInt32(txtYear.Text);
                y = new Year(e2);
                MessageBox.Show(y.check());
            }
            catch (Exception ex)
            {
                MessageBox.Show("Exception Caught!!! " + ex.Message + " at line " +
ex.StackTrace);
            }
        }

        private void btnClose_Click(object sender, EventArgs e)
        {
            this.Close();
        }
    }
}
```

```

class Year
{
    int y;

    public Year()
    {

    }

    ~Year()
    {
    }

    public Year(int e)
    {
        this.y = e;
    }

    public int getY()
    {
        return y;
    }

    public string check()
    {
        if (((y % 4 == 0) && (y % 100 != 0)) || (y % 400 == 0)) {
            return "It is Leap Year";
        }
        return "It is Not Leap Year";
    }
}

```

Program.cs

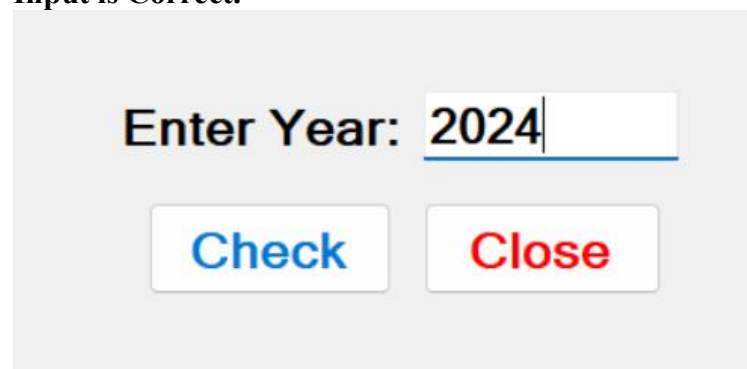
```

static void Main()
{
    Application.EnableVisualStyles();
    Application.SetCompatibleTextRenderingDefault(false);
    Application.Run(new Pract1Q2());
}

```

Output:

Input is Correct.



A screenshot of a Windows-style dialog box. It has a title bar with a close button (X). The main text reads "Enter Year: 2024" where "2024" is entered in a text input field. Below the text are two buttons: "Check" in blue text and "Close" in red text.



Input is Wrong.

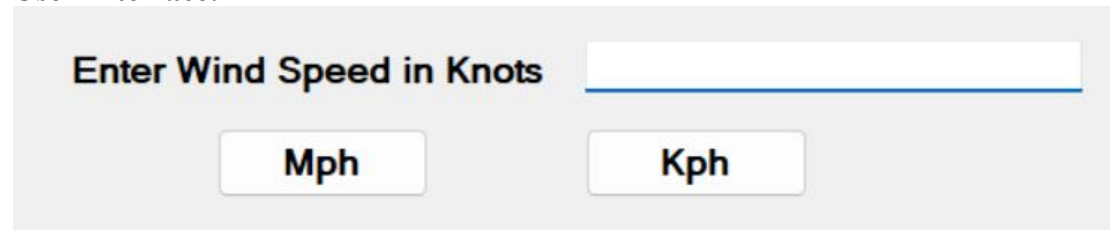
Enter Year:

×

It is Not Leap Year

Q.3 Design a Window application in c# using objects and classes for wind conversion from knots to mph, kph.

User Interface:



The user interface is a light gray rectangular window. At the top, it contains the text "Enter Wind Speed in Knots" in a bold, black, sans-serif font. To the right of this text is a white text input field with a blue border. Below the input field, there are two buttons: "Mph" on the left and "Kph" on the right. Both buttons are white with a gray border and a slight shadow effect.

Program.cs

```
static void Main()
{
    Application.EnableVisualStyles();
    Application.SetCompatibleTextRenderingDefault(false);
    Application.Run(new Pract1Q3());
}
```

PractNo1Q3.cs

```
using System.Windows.Forms;

namespace PracticalNo1All
{
    public partial class Pract1Q3 : Form
    {
        Wind d = null;
        public Pract1Q3()
        {
            InitializeComponent();
        }

        private void btnMPH_Click(object sender, EventArgs e)
        {
            double kn = Convert.ToDouble(txt1.Text);
            d = new Wind(kn);
            MessageBox.Show("Value of MPH:" + d.mph());
        }

        private void btnKPH_Click(object sender, EventArgs e)
        {
            double kn = Convert.ToDouble(txt1.Text);
            d = new Wind(kn);
            MessageBox.Show("Value of KPH:" + d.Kph());
        }
    }

    class Wind
    {
        double w;

        public Wind()
        {
        }

        public Wind(double p)
```

```

    {
        w = p;
    }

    ~Wind()
    {

    }

    public double getW()
    {
        return w;
    }

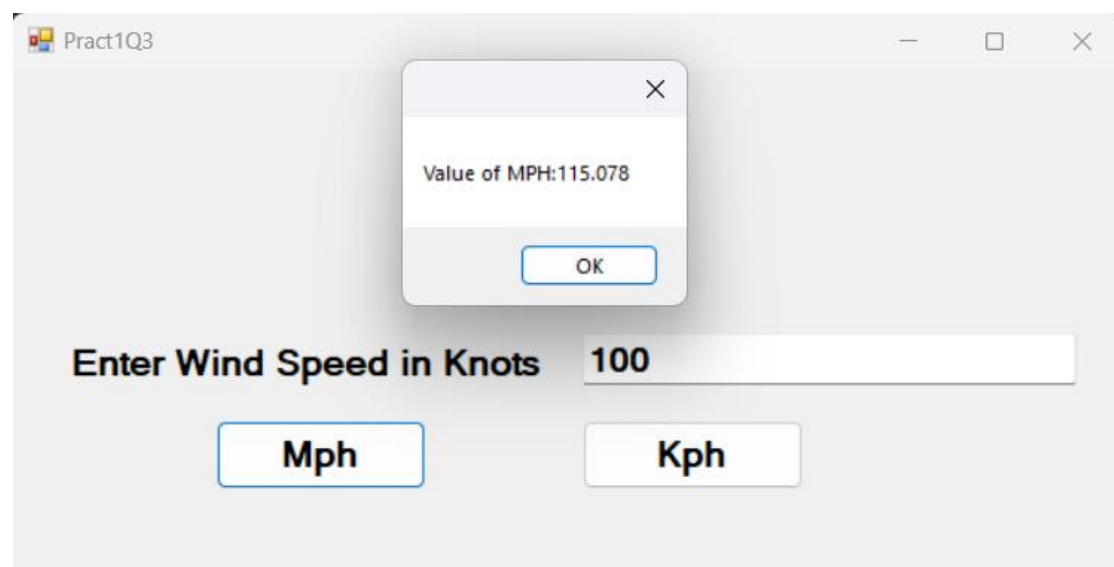
    public double mph()
    {
        return 1.15078 * w;
    }

    public double Kph()
    {
        return w * 1.852;
    }
}
}

```

Output:

Knots to MPH:

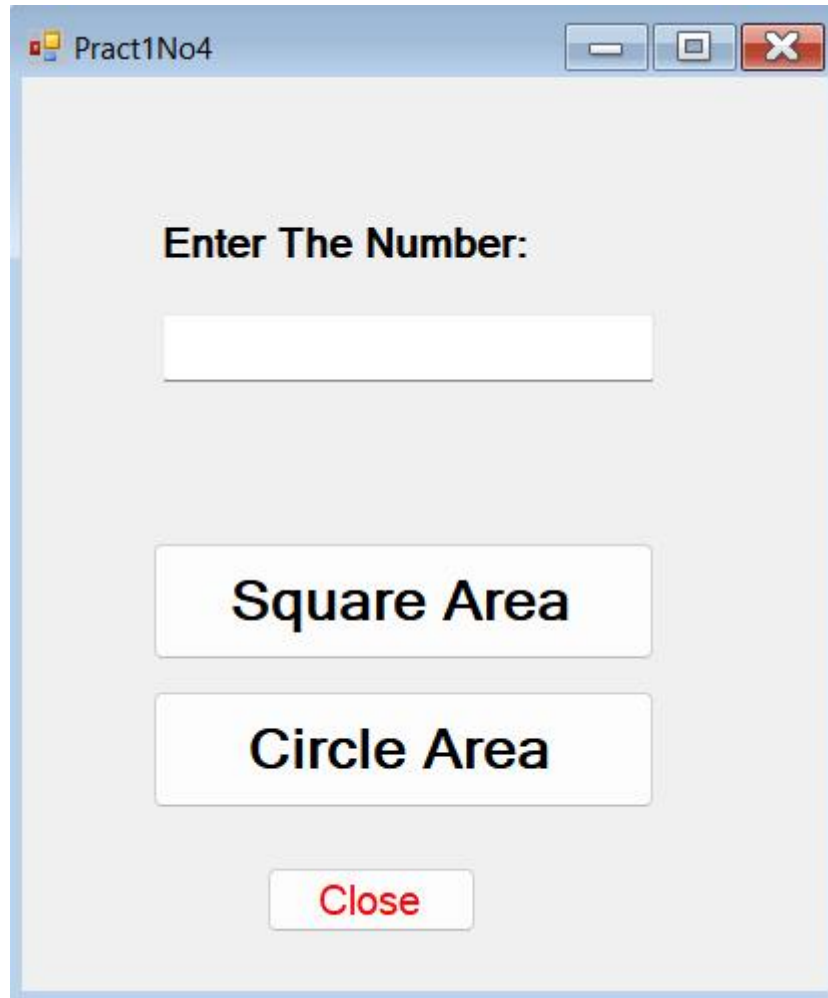


Knots to KPH:

The screenshot shows a Windows application window titled "Pract1Q3". In the center, a modal dialog box is open, displaying the text "Value of KPH:185.2" and an "OK" button. Below the dialog, the main window contains the text "Enter Wind Speed in Knots" followed by a text input field containing the number "100". At the bottom of the window, there are two buttons: "Mph" and "Kph". The "Kph" button is highlighted with a blue border, indicating it is the selected unit for conversion.

Q.4 Design a Window application to demonstrate multiple inheritance.

User Interface:



Program.cs

```
static void Main()
{
    Application.EnableVisualStyles();
    Application.SetCompatibleTextRenderingDefault(false);
    Application.Run(new Pract1No4());
}
```

PractNo1Q4.cs

```
using System.Windows.Forms;

namespace PracticalNo1All
{
    public partial class Pract1No4 : Form
    {
        public Pract1No4()
        {
            InitializeComponent();
        }

        private void baseClass_Click(object sender, EventArgs e)
        {
            double s1 = Convert.ToDouble(textBox1.Text);
            Shape2 s = new Shape2();
        }
    }
}
```

```

        MessageBox.Show("This is " + s.getArea(s1).ToString());
    }

    private void derClass_Click(object sender, EventArgs e)
    {
        double s1 = Convert.ToDouble(textBox1.Text);
        Shape2 s = new Shape2(s1);
        MessageBox.Show("This is " + s.AreaSqr().ToString());
    }

    private void btnClose_Click(object sender, EventArgs e)
    {
        this.Close();
    }
}

public interface circle
{
    double getArea(double s);
}

public interface square
{
    double AreaSqr();
}

class Shape2 : circle, square
{
    double size;

    public Shape2(double size)
    {
        this.size = size;
    }

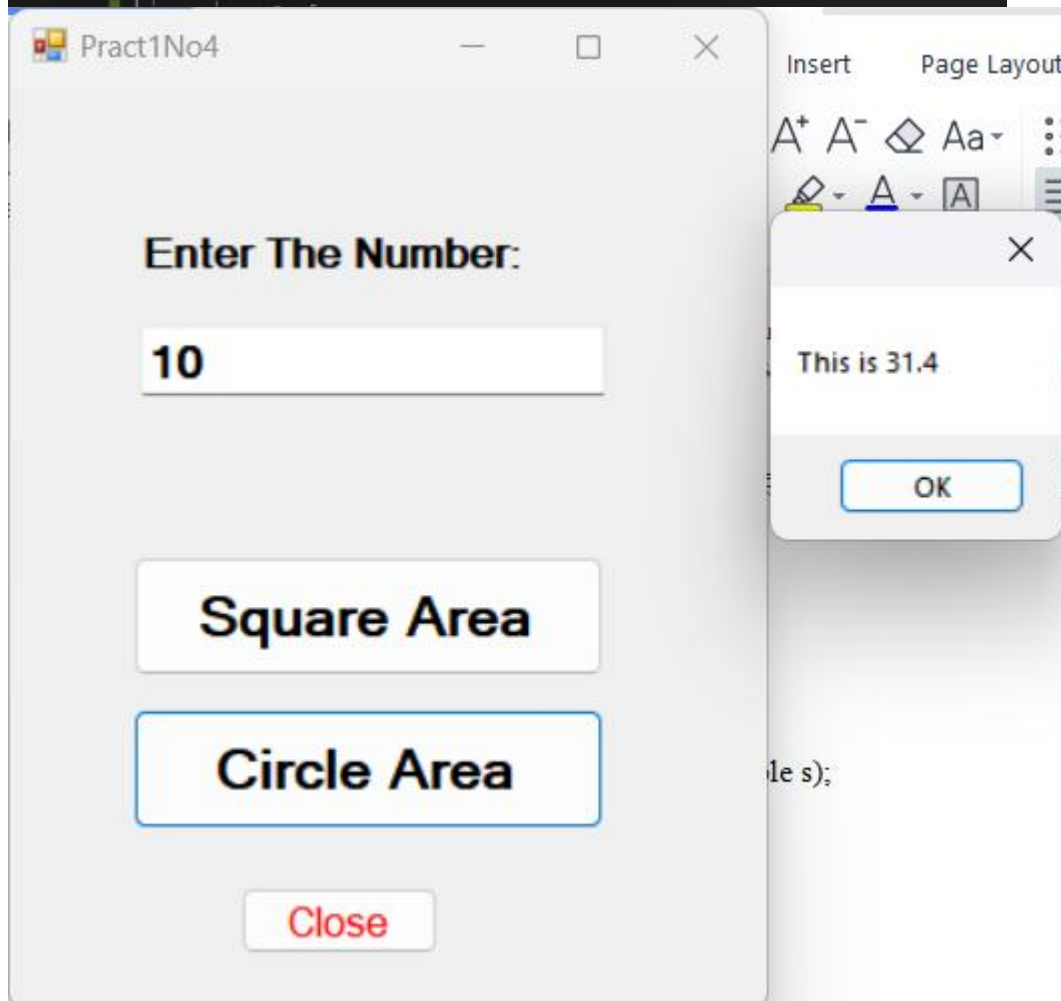
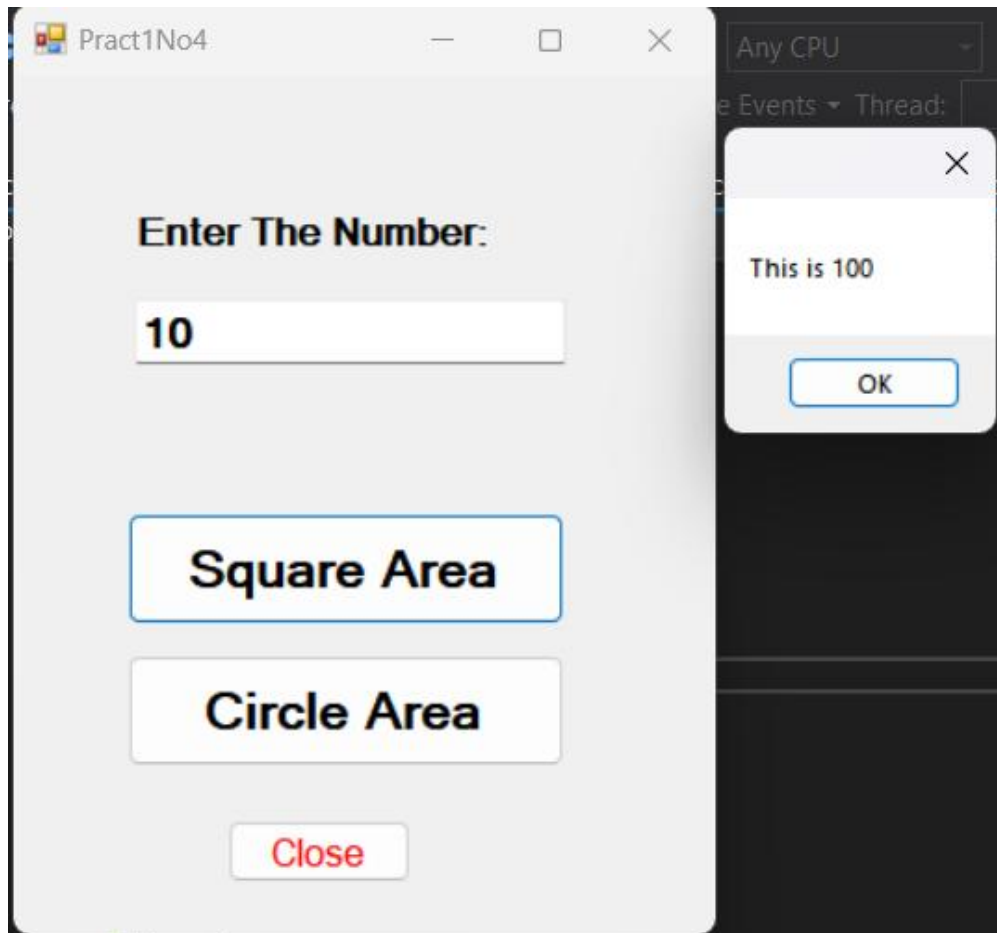
    public Shape2() { }

    public double getArea(double size)
    {
        this.size = size;
        return size * 3.14;
    }

    public double AreaSqr()
    {
        return size * size;
    }
}
}

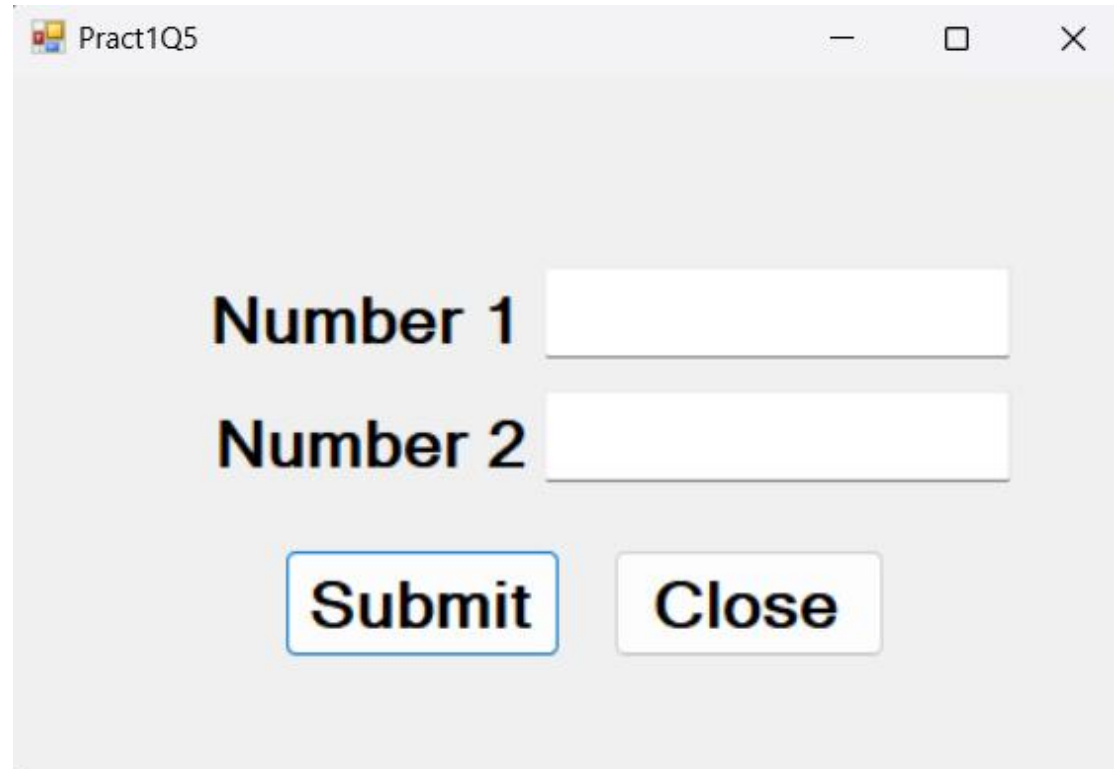
```

Output:



Q.5 Design a Window application to demonstrate abstract class.

User Interface:



Program.cs

```
static void Main()
{
    Application.EnableVisualStyles();
    Application.SetCompatibleTextRenderingDefault(false);
    Application.Run(new Pract1Q5());
}
```

PractNo1Q5.cs

```
using System.Windows.Forms;

namespace PracticalNo1All
{
    public partial class Pract1Q5 : Form
    {
        public Pract1Q5()
        {
            InitializeComponent();
        }

        private void btnSubmit_Click(object sender, EventArgs e)
        {
            int a = Convert.ToInt32(txtLength.Text);
            int b = Convert.ToInt32(txtBreadth.Text);

            Add ad = new Add(a,b);
            Sub sb = new Sub(a,b);
            Multi ml = new Multi(a,b);
            Division dv = new Division(a,b);

            MessageBox.Show("Add:" + ad.sol() + ", Subtract: " + sb.sol()+",
            Multiplication: " +ml.sol()+", Division: "+dv.sol());
        }
    }
}
```

```
private void btnClose_Click(object sender, EventArgs e)
{
    this.Close();
}
}
```

```
abstract class Calculation
{
    public virtual int sol() {
        return 0;
    }
}
```

```
class Multi : Calculation
{
    private int a;
    private int b;
    public Multi(int a, int b)
    {
        this.a = a;
        this.b = b;
    }
    public override int sol()
    {
        return a*b;
    }
}
```

```
class Division : Calculation
{
    private int l;
    private int w;
    public Division(int a, int b)
    {
        l = a;
        w = b;
    }
    public override int sol()
    {
        return l/w;
    }
}
```

```
class Add : Calculation
{
    private int l;
    private int w;
    public Add(int a, int b)
    {
        l = a;
        w = b;
    }
    public override int sol()
    {
        return l + w;
    }
}
```

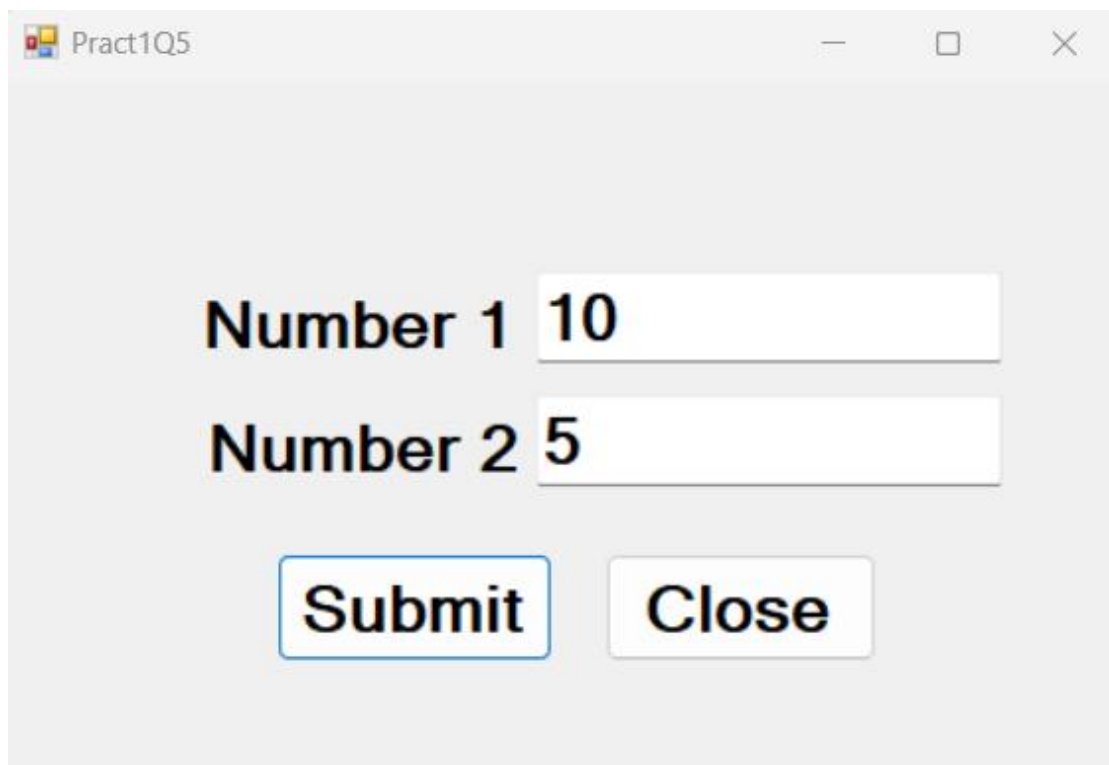
```

    }

    class Sub : Calculation
    {
        private int l;
        private int w;
        public Sub(int a, int b)
        {
            l = a;
            w = b;
        }
        public override int sol()
        {
            return l - w;
        }
    }
}

```

Output:



Pract1Q5

Number 1 10

Number 2 5

Submit Close



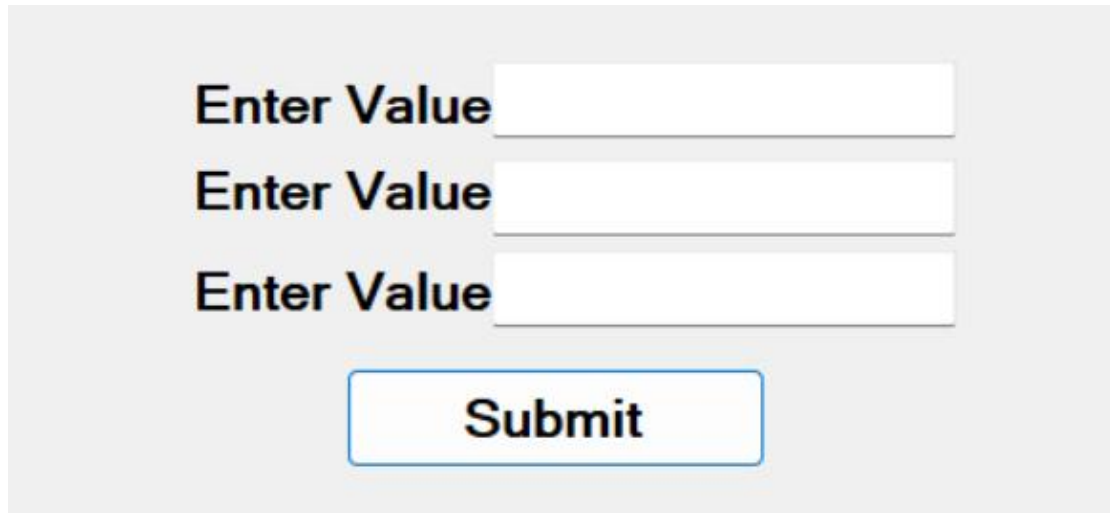
×

Add:15, Subtract: 5, Multiplication: 50, Division: 2

OK

Q.6 Design a Window application to demonstrate Indexer.

User Interface:



Enter Value

Enter Value

Enter Value

Program.cs

```
static void Main()
{
    Application.EnableVisualStyles();
    Application.SetCompatibleTextRenderingDefault(false);
    Application.Run(new Pract1Q6());
}
```

PractNo1Q6.cs

```
using System.Windows.Forms;

namespace PracticalNo1All
{
    public partial class Pract1Q6 : Form
    {
        IndexerClass team = null;
        public Pract1Q6()
        {
            InitializeComponent();
        }

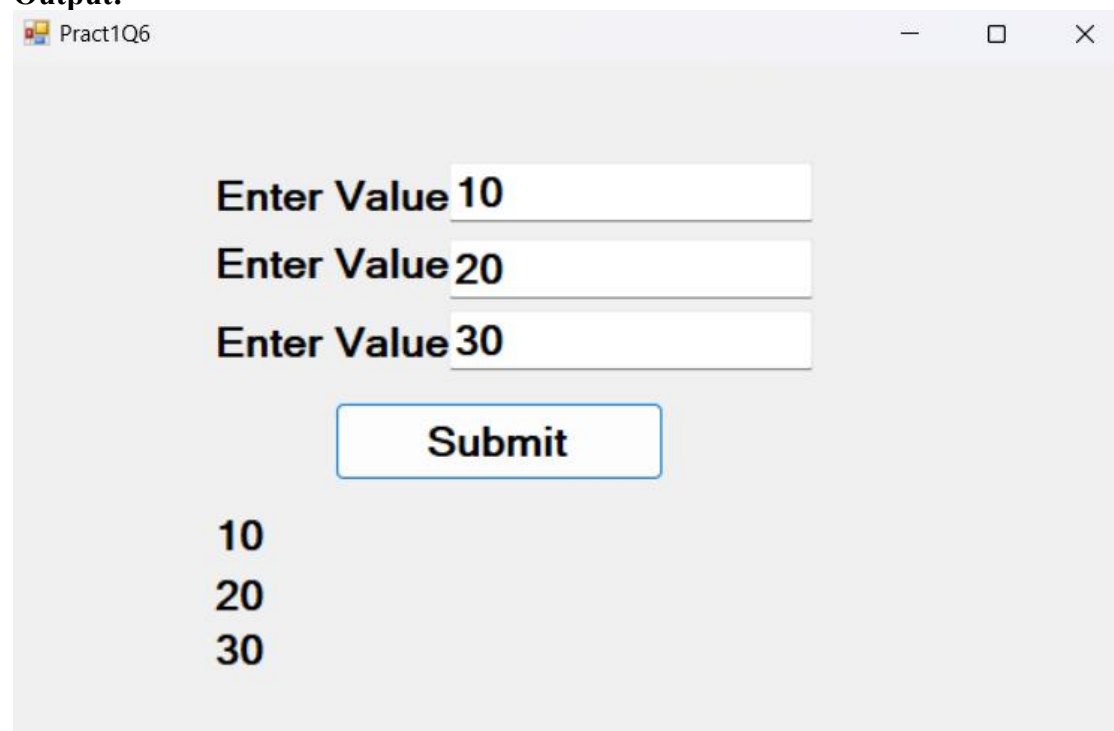
        private void button1_Click(object sender, EventArgs e)
        {
            try
            {
                team = new IndexerClass();
                team[0] = textBox1.Text;
                team[1] = textBox2.Text;
                team[2] = textBox3.Text;

                label4.Text = team[0];
                label5.Text = team[1];
                label6.Text = team[2];
            }
            catch (Exception ex)
            {
                MessageBox.Show("Error : " + ex);
            }
        }
    }
}
```



```
class IndexerClass
{
    private string[] names = new string[3];
    public string this[int i]
    {
        get
        {
            return names[i];
        }
        set
        {
            names[i] = value;
        }
    }
}
```

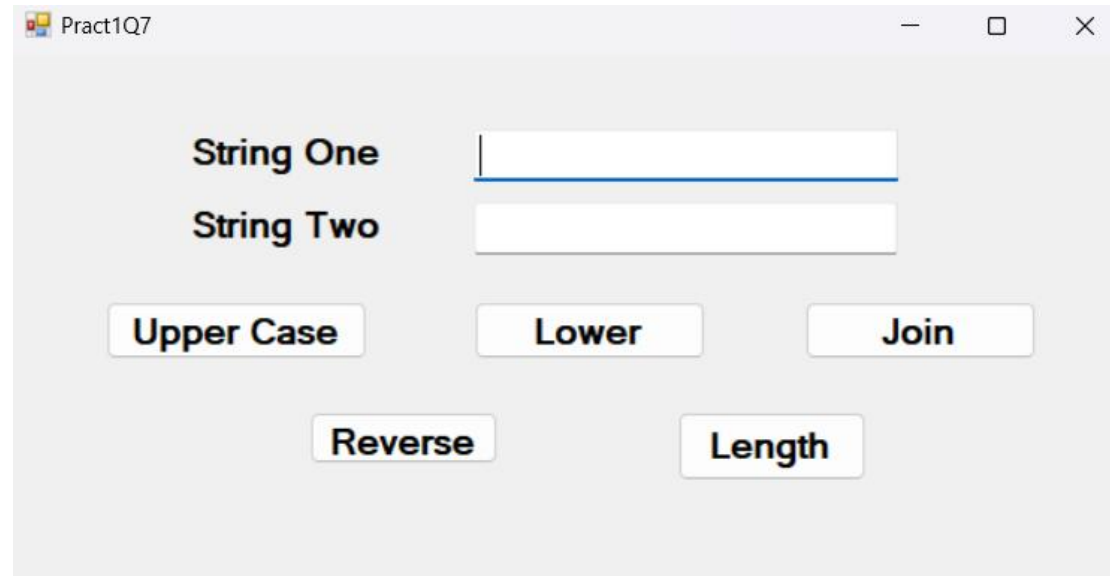
Output:



The screenshot shows a Windows application window titled "Pract1Q6". Inside the window, there is a form with three input fields. Each field is preceded by the text "Enter Value". The first field contains the number "10", the second contains "20", and the third contains "30". Below these fields is a button labeled "Submit". At the bottom left of the window, the values "10", "20", and "30" are displayed vertically.

Q.7 Design a Window application to demonstrate string manipulations.

User Interface:



The screenshot shows a Windows application window titled "Pract1Q7". Inside the window, there are two text input fields. The first field is labeled "String One" and the second is labeled "String Two". Below these fields, there are five buttons arranged in two rows. The first row contains "Upper Case", "Lower", and "Join". The second row contains "Reverse" and "Length".

Program.cs

```
static void Main()
{
    Application.EnableVisualStyles();
    Application.SetCompatibleTextRenderingDefault(false);
    Application.Run(new Pract1Q7());
}
```

PractNo1Q7.cs

```
using System.Windows.Forms;

namespace PracticalNo1All
{
    public partial class Pract1Q7 : Form
    {
        public Pract1Q7()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            string s1 = textBox1.Text;
            string s2 = textBox2.Text;
            string p = s1.ToUpper();
            string q = s2.ToUpper();
            label3.Text = p;
            label4.Text = q;
        }

        private void button2_Click(object sender, EventArgs e)
        {
            string s1 = textBox1.Text;
            string s2 = textBox2.Text;
            string p = s1.ToLower();
            string q = s2.ToLower();
            label3.Text = p;
        }
    }
}
```

```

        label4.Text = q;
    }

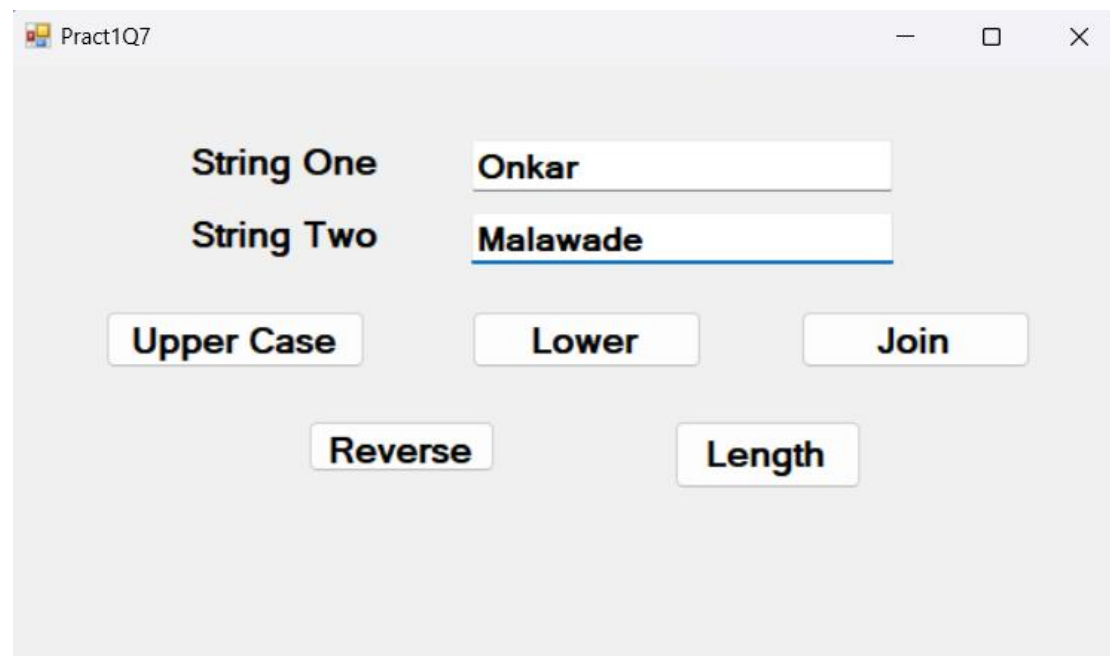
    private void button3_Click(object sender, EventArgs e)
    {
        string s1 = textBox1.Text;
        string s2 = textBox2.Text;
        MessageBox.Show(s1 + " " + s2);
    }

    private void button4_Click(object sender, EventArgs e)
    {
        string s1 = textBox1.Text;
        string s2 = textBox2.Text;
        string reversed = new string(s1.Reverse().ToArray());
        string revered2 = new string(s2.Reverse().ToArray());
        MessageBox.Show(reversed + " "+revered2);
    }

    private void button5_Click(object sender, EventArgs e)
    {
        string s1 = textBox1.Text;
        string s2 = textBox2.Text;
        MessageBox.Show("String1 Length: "+s1.Length + ", String2 Length: " +
s2.Length);
    }
}
}
}

```

Output:



Pract1Q7

String One

String Two

Upper Case Lower Join

Reverse Length

ONKAR
MALAWADE

Pract1Q7

String One

String Two

Upper Case **Lower** Join

Reverse Length

onkar
malawade

Pract1Q7

String One

String Two

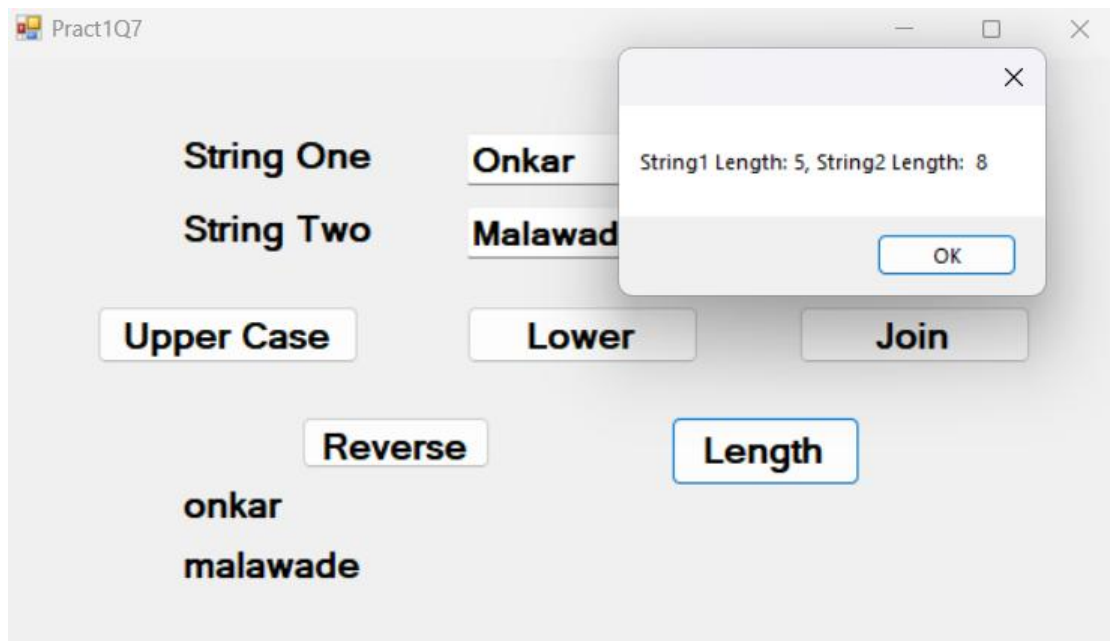
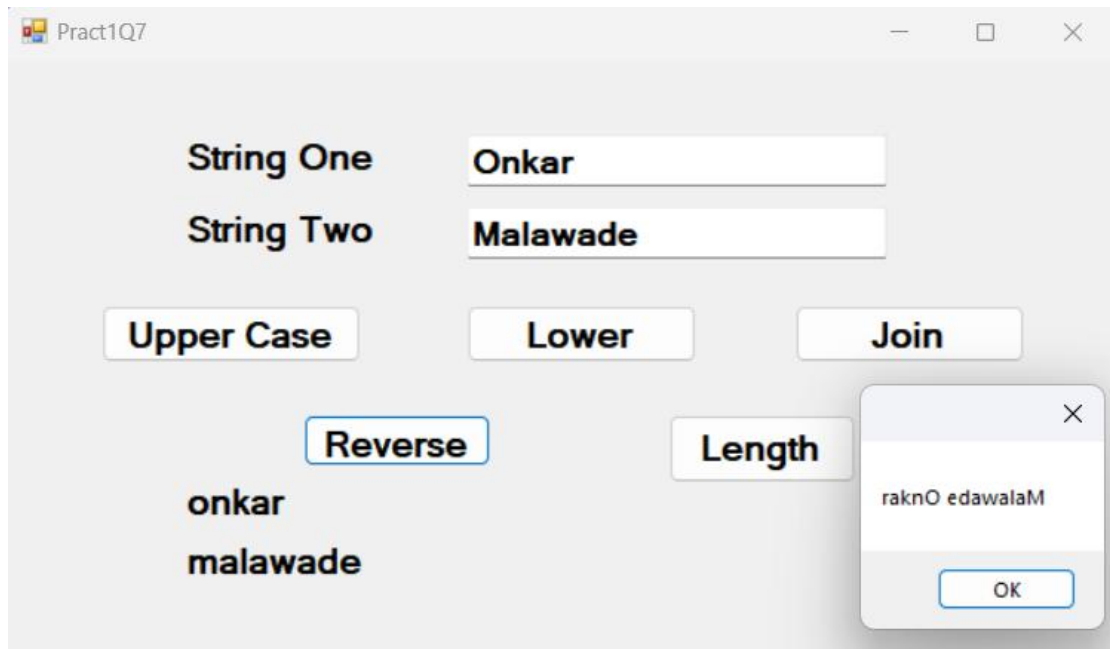
Upper Case Lower **Join**

Reverse Length

onkar
malawade

Onkar Malawade

OK



Q.8 Design a window application to show following output.

User Interface:

Pract1Q8

| | |
|-----------------------|-----------------------------------|
| Full date and time | Friday, March 1, 2024 11:05:14 PM |
| Long Date | Friday, March 1, 2024 |
| Short Date | 3/1/2024 |
| General Date and Time | 3/1/2024 11:05:14 PM |
| Long Time | 11:05:14 PM |
| Short Time | 11:05 PM |
| Days until new year | 305 |

Refresh Clear

Program.cs

```
static void Main()
{
    Application.EnableVisualStyles();
    Application.SetCompatibleTextRenderingDefault(false);
    Application.Run(new Pract1Q8());
}
```

PractNo1Q8.cs

```
using System.Windows.Forms;

namespace PracticalNo1All
{
    public partial class Pract1Q8 : Form
    {
        DateTime date = DateTime.Now;
        public Pract1Q8()
        {
            InitializeComponent();
        }

        private void button2_Click(object sender, EventArgs e)
        {
            this.Close();
        }

        private void button1_Click(object sender, EventArgs e)
        {
            this.Refresh();
        }

        private void Pract1Q8_Load(object sender, EventArgs e)
        {
            textBox1.Text = date.ToString("F");
            textBox2.Text = date.ToString("D");
        }
    }
}
```

```
        textBox3.Text = date.ToString("d");
        textBox4.Text = date.ToString("G");
        textBox5.Text = date.ToString("T");
        textBox6.Text = date.ToString("t");
        int daysInYear = DateTime.IsLeapYear(date.Year) ? 365 : 366;
        textBox7.Text = (daysInYear - date.DayOfYear).ToString();
    }
}
}
```