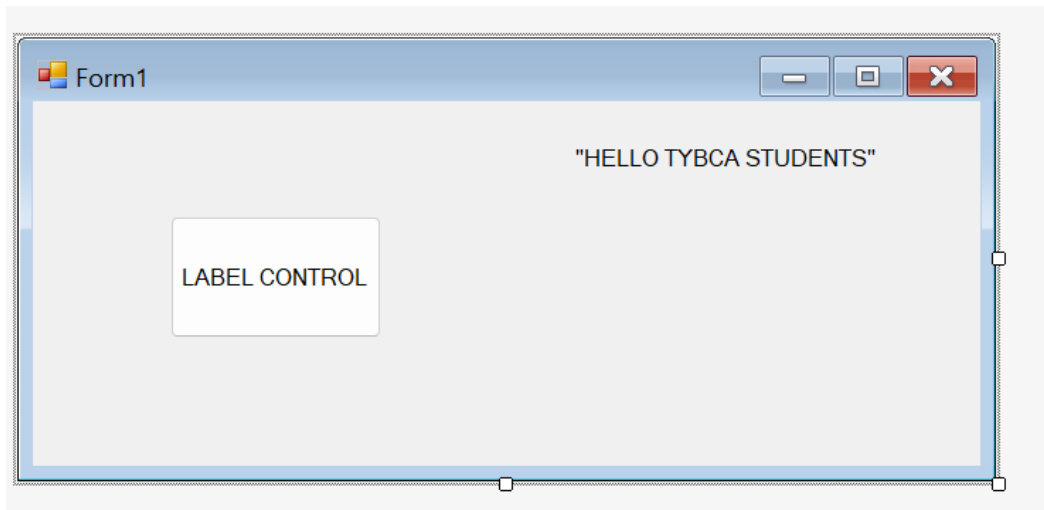


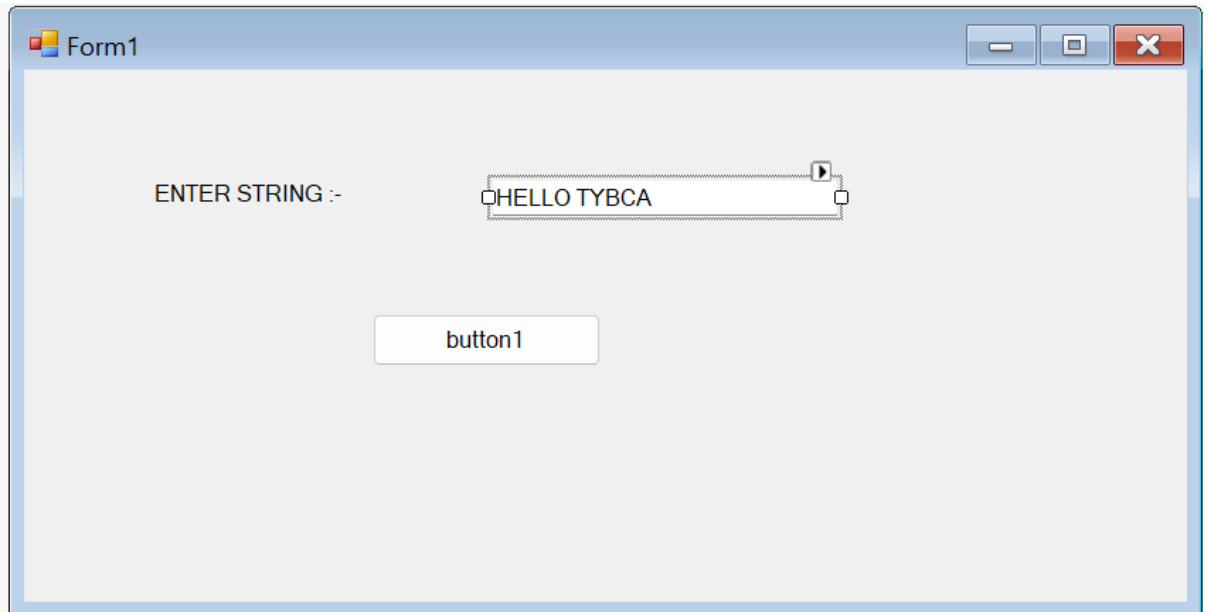
## 1) Label Control using .NET framework Windows Application.



### Coding part on button's clicks: -

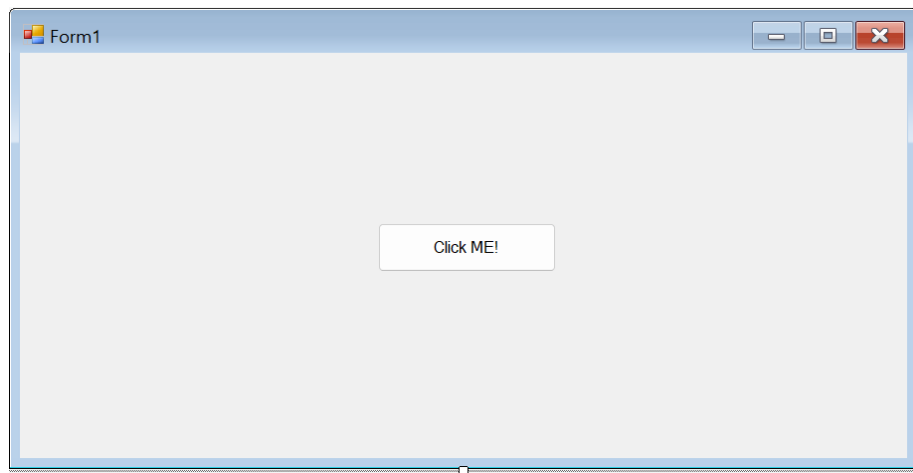
```
private void button1_Click(object sender, EventArgs e)
{
    //label1.BackColor = Color.Wheat;
    // label1.ForeColor = Color.Red;
    //label1.Text = "ROFEL";
    //label1.Font = new Font("Arial", label1.Font.Size,
label1.Font.Style);
    //label1.Font = new Font(label1.Font.Name, 20,
label1.Font.Style);
    // label1.Visible = true;
    // label1.Visible = false;
    // label1.TextAlign = ContentAlignment.TopCenter;
    // label1.Image = Image.FromFile("C:/me.jpg");
```

## 2) Textbox Control using .NET framework Windows Application.



```
private void button1_Click(object sender, EventArgs e)
{
    // MessageBox.Show(textBox1.Text);
    // int textLength = textBox1.TextLength;
    // MessageBox.Show("The length of the text is: " +
textLength.ToString());
    // textBox1.PasswordChar = '*';
    // textBox1.AppendText("New line of text.");
    // textBox1.BackColor = System.Drawing.Color.Red;
}
```

### 3) Button Control using .NET framework Windows Application.



```
private void button1_Click(object sender, EventArgs e)
{
    label1.Text = "Welcome!";
    label1.ForeColor = Color.Blue;
    label1.Font = new Font("Segoe UI", 14, FontStyle.Italic);
    label1.Location = new Point(50, 80);

    button1.Text = "Clicked!";
    button1.BackColor = Color.Orange;
    button1.Size = new Size(200, 60);
    button1.Cursor = Cursors.WaitCursor;
}

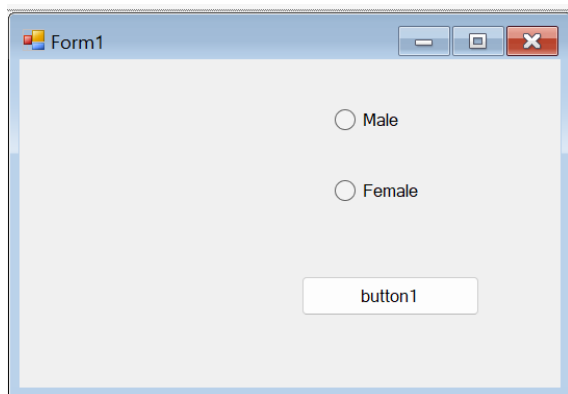
button1.Location = new Point(100, 150);
button1.TextAlign = ContentAlignment.MiddleCenter;
button1.ImageAlign = ContentAlignment.MiddleLeft;

ToolTip toolTip1 = new ToolTip();
toolTip1.SetToolTip(button1, "Click this button to continue");

button1.FlatStyle = FlatStyle.Flat;
button1.FlatAppearance.BorderSize = 2;
button1.FlatAppearance.BorderColor = Color.Red;
button1.Cursor = Cursors.Hand;

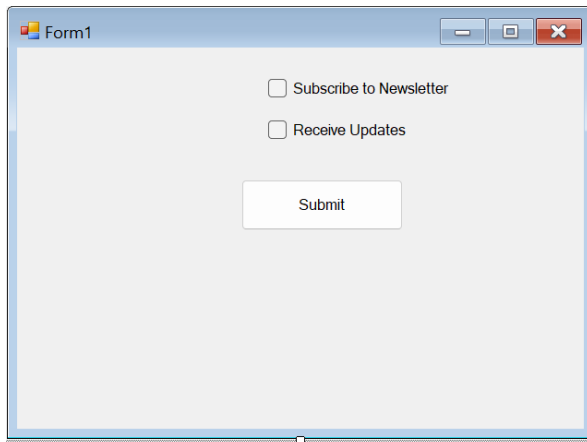
button1.Image = Image.FromFile("C:/me.jpg");
button1.ImageAlign = ContentAlignment.MiddleLeft;
```

#### 4) Radio Button Control using .NET framework Windows Application.



```
if (rbMale.Checked)
{
    MessageBox.Show("You selected Male");
}
else if (rbFemale.Checked)
{
    MessageBox.Show("You selected Female");
}
else
{
    MessageBox.Show("Please select an option.");
}
```

### 5) Check Box Control using .NET framework Windows Application.



```
string message = "You have selected:\n";

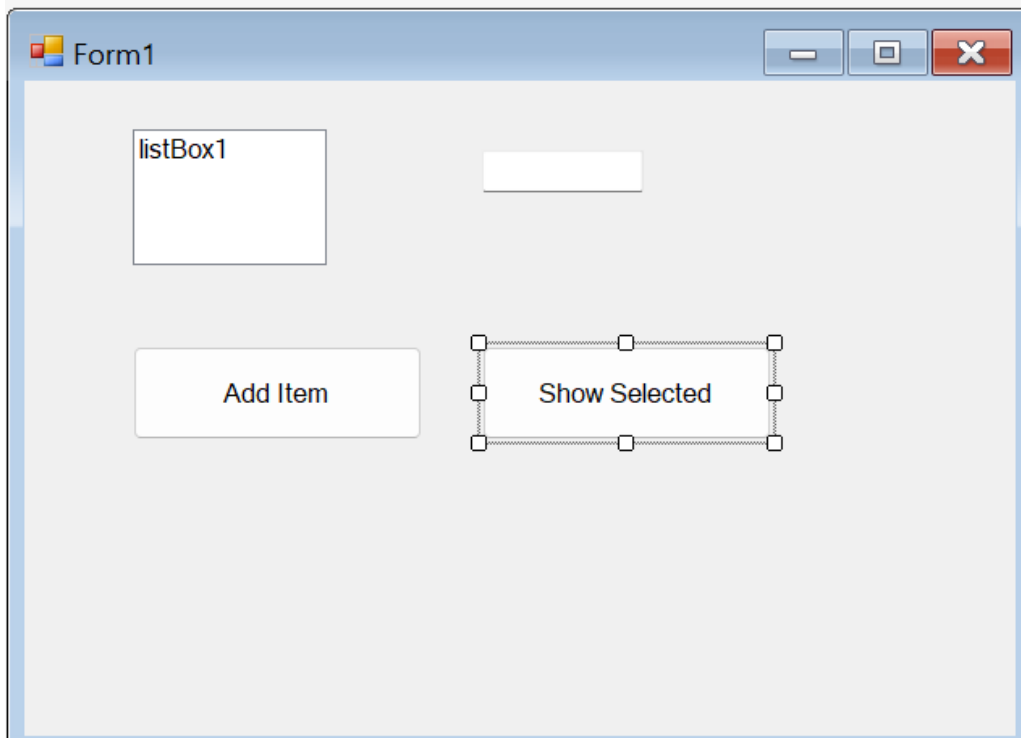
if (cbSubscribe.Checked)
    message += "- Subscribe to Newsletter\n";

if (cbUpdates.Checked)
    message += "- Receive Updates\n";

if (!cbSubscribe.Checked && !cbUpdates.Checked)
    message = "No options selected.";

MessageBox.Show(message);
```

## 6) List Box Control using .NET framework Windows Application.



### Design (Form Layout)

1. Open your Windows Forms App in Visual Studio.
2. From the Toolbox, drag and drop:
  - **ListBox** → Name: listBox1
  - **TextBox** → Name: textBox1
  - **Button** → Name: btnAdd, Text: "Add Item"
  - **Button** → Name: btnShow, Text: "Show Selected"

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

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

```
// Add item to ListBox  
private void btnAdd_Click(object sender, EventArgs e)
```

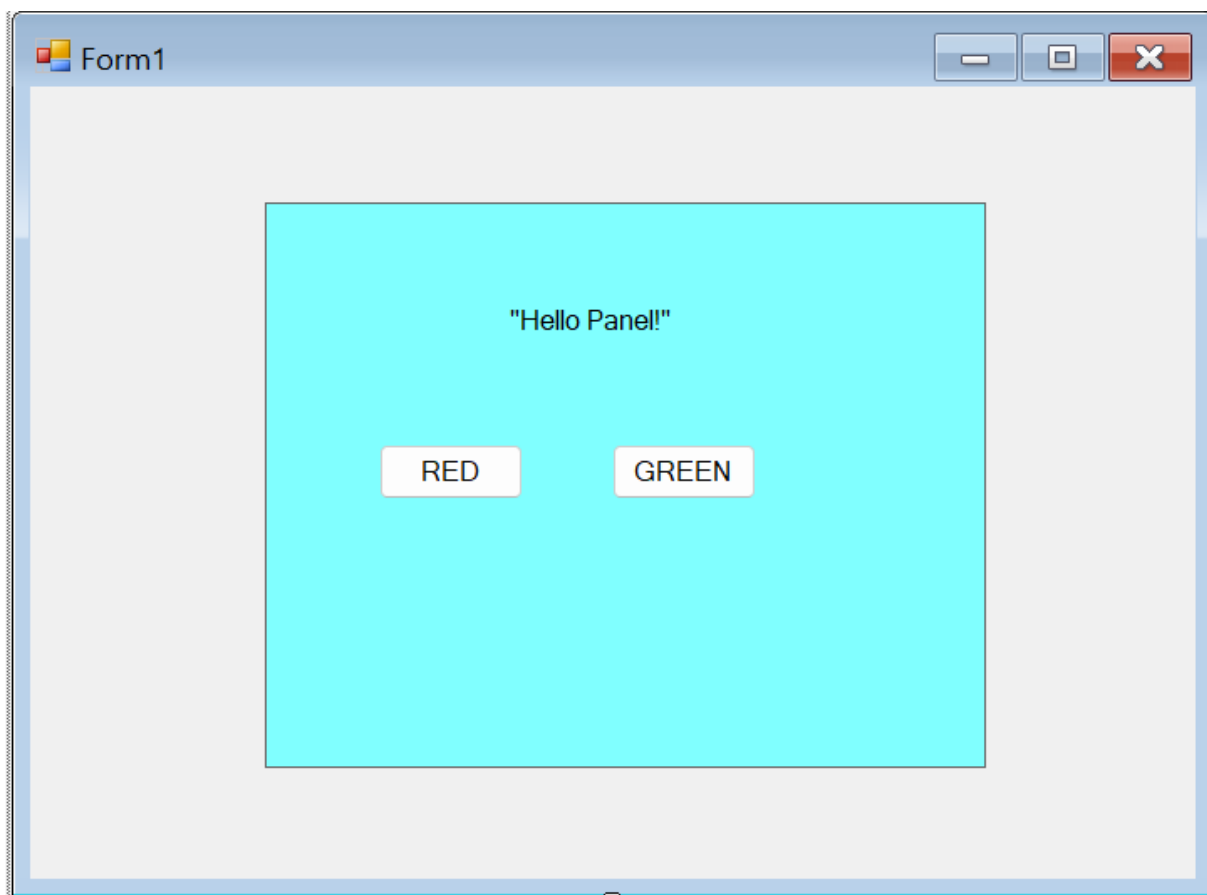
```
{
    if (!string.IsNullOrWhiteSpace(textBox1.Text))
    {
        listBox1.Items.Add(textBox1.Text);
        textBox1.Clear();
    }
    else
    {
        MessageBox.Show("Please enter an item name.");
    }
}

// Show selected item
private void btnShow_Click(object sender, EventArgs e)
{
    if (listBox1.SelectedItem != null)
    {
        MessageBox.Show("You          selected:          "          +
listBox1.SelectedItem.ToString());
    }
    else
    {
        MessageBox.Show("Please select an item from the list.");
    }
}
}
```

## 7) Panel Control using .NET framework Windows Application.

### Design Part (Toolbox Placement)

1. Open your Windows Forms Application.
2. From Toolbox, drag and drop a Panel control on the form.
  - Name: panel1
  - Set BackColor = LightBlue
  - Set BorderStyle = FixedSingle
3. Drag and drop two Button controls inside the panel.
  - Name: btnRed → Text = "Red"
  - Name: btnGreen → Text = "Green"
4. Add one Label inside the panel.
  - Name: label1 → Text = "Hello Panel!"





**Coding Part: -**

```
private void btnred_Click(object sender, EventArgs e)
{
    panel1.BackColor = System.Drawing.Color.Red;
    label1.Text = "Panel turned RED!";
}

private void btngreen_Click(object sender, EventArgs e)
{
    panel1.BackColor = System.Drawing.Color.Green;
    label1.Text = "Panel turned GREEN!";
}
```

**8) Panel Control using .NET framework Windows Application.  
Runtime example (everything created through code, no Toolbox needed).**

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

namespace PanelExampleRuntime
{
    public partial class Form1 : Form
    {
        private Panel panel1;
        private Button btnRed;
        private Button btnGreen;
        private Label label1;

        public Form1()
        {
            InitializeComponent();
            CreatePanelExample();
        }

        private void CreatePanelExample()
        {
            // Panel
            panel1 = new Panel();
            panel1.Size = new System.Drawing.Size(300, 200);
            panel1.Location = new System.Drawing.Point(30, 30);
            panel1.BackColor = System.Drawing.Color.LightBlue;
            panel1.BorderStyle = BorderStyle.FixedSingle;

            // Label
            label1 = new Label();
            label1.Text = "Hello Panel!";
            label1.Location = new System.Drawing.Point(100, 20);
            label1.AutoSize = true;

            // Button Red
            btnRed = new Button();
            btnRed.Text = "Red";
            btnRed.Location = new System.Drawing.Point(50, 100);
```

```
        btnRed.Click += BtnRed_Click;

        // Button Green
        btnGreen = new Button();
        btnGreen.Text = "Green";
        btnGreen.Location = new System.Drawing.Point(150, 100);
        btnGreen.Click += BtnGreen_Click;

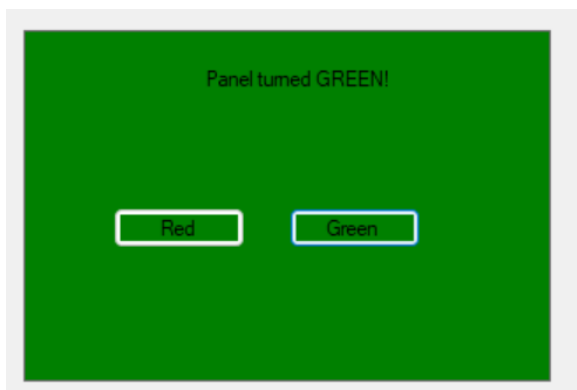
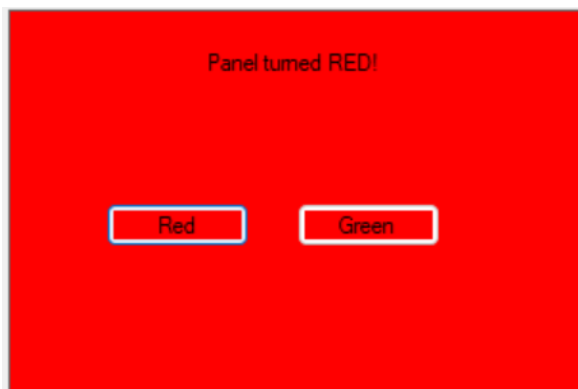
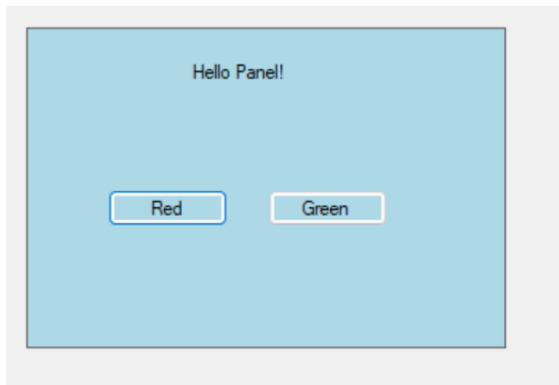
        // Add controls to Panel
        panel1.Controls.Add(label1);
        panel1.Controls.Add(btnRed);
        panel1.Controls.Add(btnGreen);

        // Add Panel to Form
        this.Controls.Add(panel1);
    }

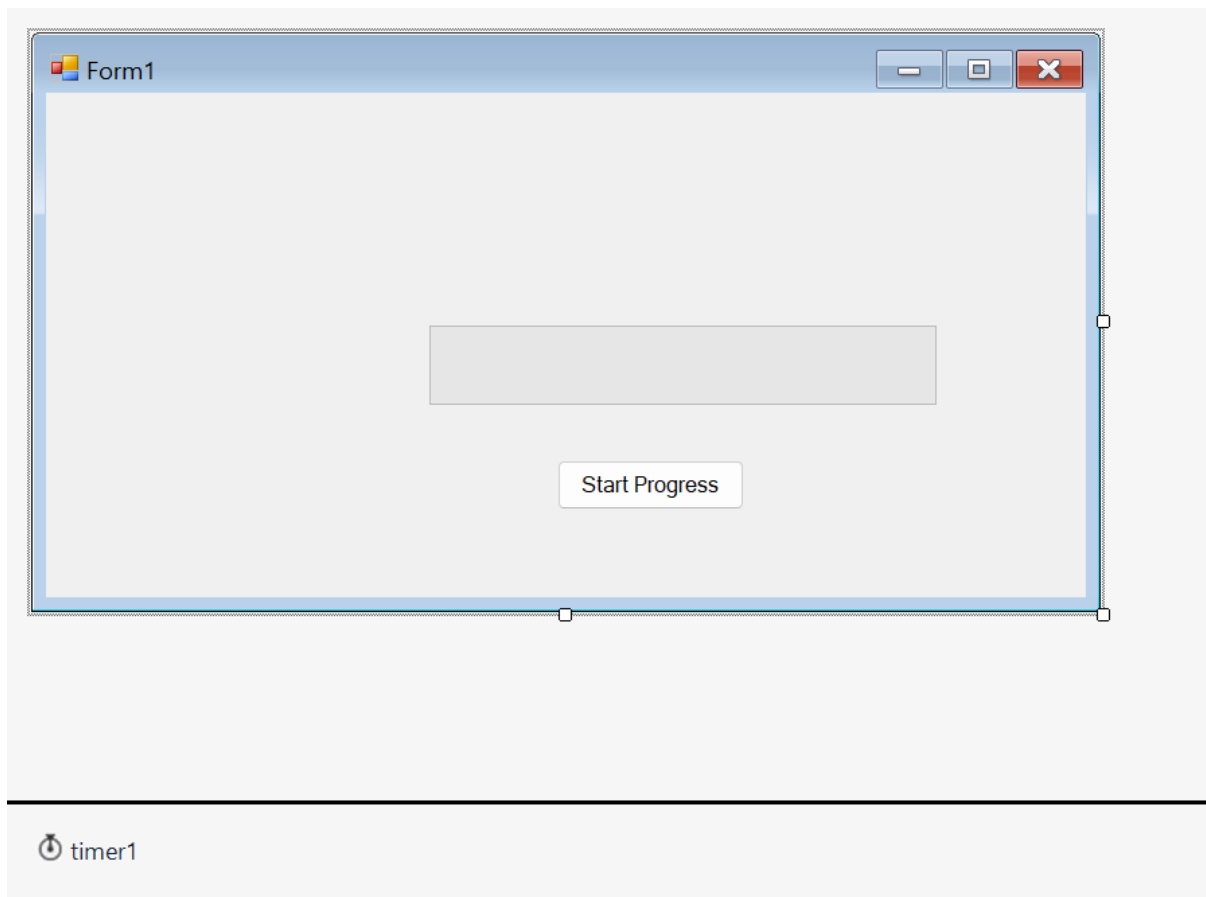
    private void BtnRed_Click(object sender, EventArgs e)
    {
        panel1.BackColor = System.Drawing.Color.Red;
        label1.Text = "Panel turned RED!";
    }

    private void BtnGreen_Click(object sender, EventArgs e)
    {
        panel1.BackColor = System.Drawing.Color.Green;
        label1.Text = "Panel turned GREEN!";
    }
}
```

## OUTPUT: -



## 10) Progress Bar Control using .NET framework Windows Application.



### Coding Part :-

```
// Button Click starts progress
private void btnStart_Click(object sender, EventArgs e)
{
    progressBar1.Value = 0; // reset
    timer1.Start();        // start timer
}

// Timer Tick event updates ProgressBar
private void timer1_Tick(object sender, EventArgs e)
{

```

```
        if (progressBar1.Value < progressBar1.Maximum)
        {
            progressBar1.PerformStep(); // increase progress
        }
        else
        {
            timer1.Stop();           // stop when complete
            MessageBox.Show("Task Completed!");
        }
    }
}
```

## OUTPUT :-

