



LABTASK WEEK-10

ALI ZIA KHAN
FESE-19052

Create a simple C# that will place one combo box filled with the names of different shapes by using item property of combo box and application should be able to draw the selected shape of combo box control by creating graphic ,pen and solid brush objects.

```
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 lab_10_Q1_combobox_graphic
{
    3 references
    public partial class Form1 : Form
    {
        1 reference
        public Form1()
        {
            InitializeComponent();
        }

        1 reference
        private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
        {
            Graphics g = base.CreateGraphics();
            SolidBrush sb1 = new SolidBrush(Color.Aqua);
            SolidBrush sb2 = new SolidBrush(Color.DarkGray);
            SolidBrush sb3 = new SolidBrush(Color.DarkRed);
            Pen p1 = new Pen(Color.Coral);
            Pen p2 = new Pen(Color.BlueViolet);
            Pen p3 = new Pen(Color.Pink);
        }
    }
}
```

Task 2:

Create an array of Buttons that would display buttons dynamically at run time.

```
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;

namespace LAB10_Q2
{
    3 references
    public partial class Form1 : Form
    {
        1 reference
        public Form1()
        {
            InitializeComponent();
        }

        1 reference
        private void Form1_Load(object sender, EventArgs e)
        {
            Button[] p = new Button[5];

            for (int i = 1; i <= b.Length - 1; i++)
            {
                ...
            }
        }
    }
}
```

1 reference

```
private void Form1_Load(object sender, EventArgs e)
{
    Button[] p = new Button[5];

    for (int i = 1; i <= p.Length - 1; i++)
    {
        p[i] = new Button();
        p[i].Text = "LBTSK-10";
        p[i].Size = new Size(200, 60);
        p[i].Location = new Point(i + 70, i + 100);
        p[i].Left = 50 * i;
        p[i].Top = i * 70;
        p[i].BackColor = Color.Black;
        p[i].ForeColor = Color.Beige;
        p[i].Font = new Font("Arial", 24, FontStyle.Bold);

        this.Controls.Add(p[i]);
    }
}
}
```

Create a slideshow of pictures by using picture box control and timer control. Make a folder on any drive containing all of your pictures, you want to include in a slideshow. Rename all the pictures starting with 1.jpg, 2.jpg... n.jpg.

```
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 lab_10_Q3_picslideshow
{
    3 references
    public partial class Form1 : Form
    {
        1 reference
        public Form1()
        {
            InitializeComponent();
        }
        int count = 1;
        1 reference
        private void timer1_Tick(object sender, EventArgs e)
        {
            if (count == 7)
            {
                count = 1;
            }
            pictureBox1.ImageLocation = string.Format(@"D:\qwerty\{0}.jpg", count);
            count++;
        }
    }
}
```

Q: Using Mouse up, down and Mouse move events, Create an application that can draw freehand drawing.

```
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 lab_10_Q4_free_hand_drawing
{
    3 references
    public partial class Form1 : Form
    {
        1 reference
        public Form1()
        {
            InitializeComponent();
        }
        Boolean shoulddraw = true;
        1 reference
        private void Form1_Load(object sender, EventArgs e)
        {
            shoulddraw = false;
        }

        1 reference
        private void Form1_MouseDown(object sender, MouseEventArgs e)
        {
            shoulddraw = true;
        }
    }
}
```

1 reference

```
private void Form1_MouseMove(object sender, MouseEventArgs e)
```

```
{
```

```
    if (shoulddraw)
```

```
    {
```

```
        Graphics g = base.CreateGraphics();
```

```
        Random r = new Random();
```

```
        SolidBrush sb = new SolidBrush(Color.FromArgb(r.Next(0,255),r.Next(0,255),r.Next(0,255)));
```

```
        g.FillEllipse(sb, e.X, e.Y, 10, 10);
```

```
    }
```

```
}
```

1 reference

```
private void Form1_MouseUp(object sender, MouseEventArgs e)
```

```
{
```

```
    shoulddraw = false;
```

```
}
```

```
}
```

```
}
```