

Федеральное государственное бюджетное образовательное учреждение высшего образования  
«Саратовский государственный технический университет имени Гагарина Ю. А.»

Кафедра  
«Прикладные информационные технологии»

**ОТЧЕТ**  
**по лабораторной работе №7**

Студента группы 62-ПИНФ21  
Нефёдова Д. В.

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

namespace Lab_7
{
    static class Program
    {
        /// <summary>
        /// Главная точка входа для приложения.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new MainForm());
        }
    }
}
```

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

namespace Lab_7
{
    public partial class MainForm : Form
    {
        private Control selectedElement;
        private readonly ElementFactory elementFactory;

        public MainForm()
        {
            InitializeComponent();
            elementFactory = new ElementFactory();
        }

        private void ToolStripButton1_Click(object sender, EventArgs e)
        {
            AddElementToPanelAndAddHadler(elementFactory.CreateTextBox());
        }

        private void ToolStripButton2_Click(object sender, EventArgs e)
        {
            AddElementToPanelAndAddHadler(elementFactory.CreateLabel());
        }

        private void ToolStripButton3_Click(object sender, EventArgs e)
        {
            AddElementToPanelAndAddHadler(elementFactory.CreateCheckBox());
        }

        private void ToolStripButton4_Click(object sender, EventArgs e)
        {
            AddElementToPanelAndAddHadler(elementFactory.CreateComboBox());
        }

        private void ToolStripButton5_Click(object sender, EventArgs e)
        {
            AddElementToPanelAndAddHadler(elementFactory.CreateButton());
        }

        private void AddElementToPanelAndAddHadler(Control component)
        {
            component.MouseClick += new MouseEventHandler(SelectOnClick);
            panel1.Controls.Add(component);
        }
    }
}
```

```
private void SelectOnClick(object sender, MouseEventArgs e)
{
    selectedElement = sender as Control;

    textBox1.Text = selectedElement.Location.X.ToString();
    textBox2.Text = selectedElement.Location.Y.ToString();
    textBox3.Text = selectedElement.Width.ToString();
    textBox4.Text = selectedElement.Height.ToString();
    textBox5.Text = selectedElement.Text;
}

private void TextBox1_TextChanged(object sender, EventArgs e)
{
    if (ExistSelected())
        selectedElement.Location = new Point(ChangedValue(textBox1),
selectedElement.Location.Y);
}

private void TextBox2_TextChanged(object sender, EventArgs e)
{
    if (ExistSelected())
        selectedElement.Location = new Point(selectedElement.Location.X,
ChangedValue(textBox2));
}

private void TextBox3_TextChanged(object sender, EventArgs e)
{
    if (ExistSelected())
        selectedElement.Width = ChangedValue(textBox3);
}

private void TextBox4_TextChanged(object sender, EventArgs e)
{
    if (ExistSelected())
        selectedElement.Height = ChangedValue(textBox4);
}

private int ChangedValue(Control textField)
{
    if (!int.TryParse(textField.Text, out int newValue) && textField.Text != "" &&
textField.Text != "-")
    {
        textField.Text = 0.ToString();
        UncorrectMessageShow();
    }

    return newValue;
}
```

```
}

private void TextBox5_TextChanged(object sender, EventArgs e)
{
    if (ExistSelected())
        selectedElement.Text = textBox5.Text;
}

private void Button2_Click(object sender, EventArgs e)
{
    if (ExistSelected())
    {
        panel1.Controls.Remove(selectedElement);
        textBox1.Text = "";
        textBox2.Text = "";
        textBox3.Text = "";
        textBox4.Text = "";
        textBox5.Text = "";
    }
}

private bool ExistSelected()
{
    return selectedElement != null;
}

private void UncorrectMessageShow()
{
    MessageBox.Show("Некорректное значение");
}
}

class ElementFactory
{
    private readonly int defaultXCoord = 20;
    private readonly int defaultYCoord = 20;
    private readonly ElementChangeMethods methods = new ElementChangeMethods();

    public TextBox CreateTextBox()
    {
        TextBox newTextBox = new TextBox
        {
            Location = new Point(defaultXCoord, defaultYCoord)
        };

        AddEventHandlers(newTextBox);
        return newTextBox;
    }
}
```

```
public Label CreateLabel()
{
    Label newLabel = new Label
    {
        BorderStyle = BorderStyle.FixedSingle,
        Location = new Point(defaultXCoord + 30, defaultYCoord + 30),
        ForeColor = SystemColors.ControlText,
    };

    AddEventHandlers(newLabel);
    return newLabel;
}

public CheckBox CreateCheckBox()
{
    CheckBox newCheckBox = new CheckBox
    {
        Location = new Point(defaultXCoord + 60, defaultYCoord + 60)
    };

    AddEventHandlers(newCheckBox);
    return newCheckBox;
}

public ComboBox CreateComboBox()
{
    ComboBox newComboBox = new ComboBox
    {
        Location = new Point(defaultXCoord + 80, defaultYCoord + 80)
    };

    AddEventHandlers(newComboBox);
    return newComboBox;
}

public Button CreateButton()
{
    Button newButton = new Button
    {
        BackColor = Color.White,
        ForeColor = Color.Black,
        Location = new Point(defaultXCoord + 100, defaultYCoord + 100)
    };

    AddEventHandlers(newButton);
    return newButton;
}
```

```
private void AddEventHandlers(Control element)
{
    element.MouseDown += new MouseEventHandler(methods.MouseDown);
    element.MouseUp += new MouseEventHandler(methods.MouseUp);
    element.MouseMove += new MouseEventHandler(methods.MouseMove);
}

class ElementChangeMethods
{
    private Control selectedElem;
    private int deltaX;
    private int deltaY;

    public void MouseDown(object component, MouseEventArgs e)
    {
        Control element = component as Control;

        selectedElem = element;

        deltaX = Cursor.Position.X - element.Location.X;
        deltaY = Cursor.Position.Y - element.Location.Y;
    }

    public void MouseUp(object component, MouseEventArgs e)
    {
        selectedElem = null;
    }

    public void MouseMove(object component, MouseEventArgs e)
    {
        Control element = component as Control;

        if (selectedElem == element)
        {
            element.Location = new Point(
                Cursor.Position.X - deltaX,
                Cursor.Position.Y - deltaY
            );
        }
    }
}
```

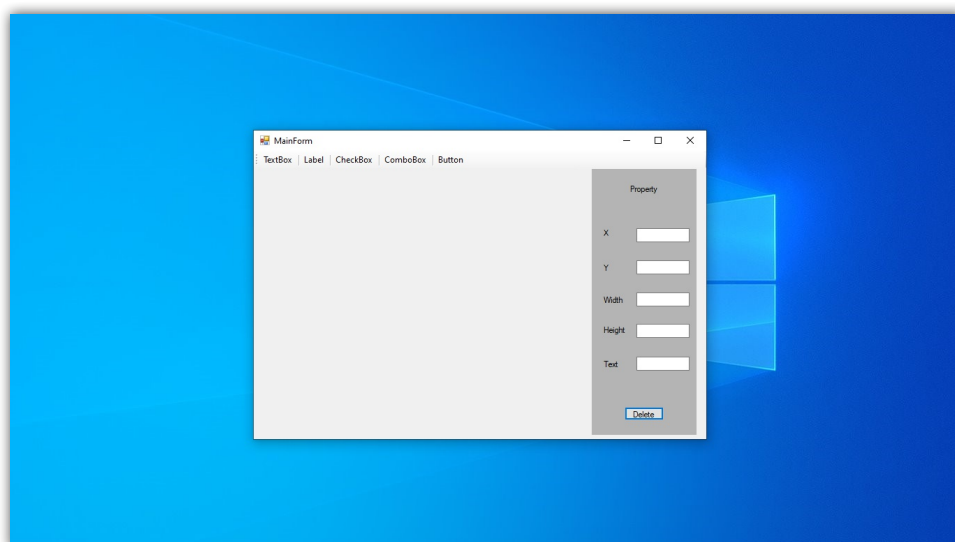


Рис. 1. Приложение сразу после запуска.

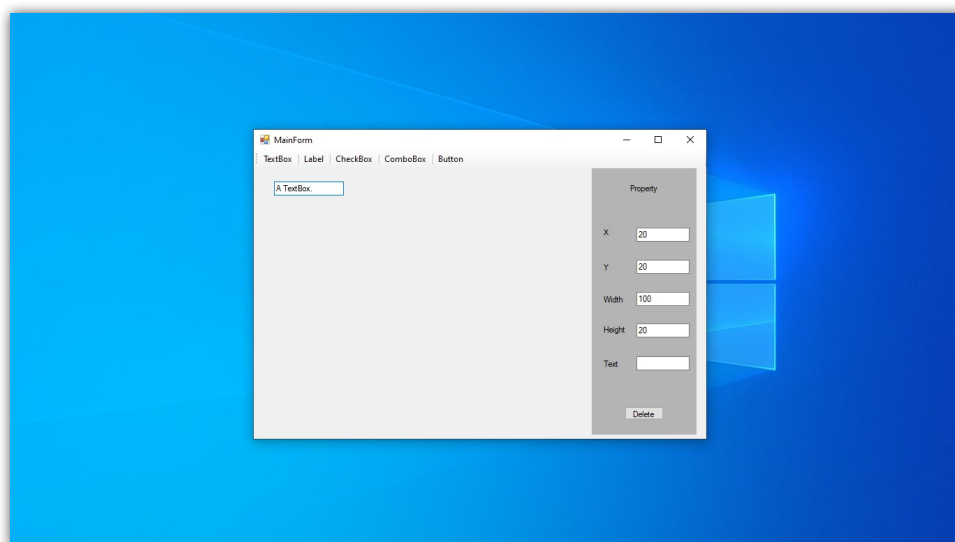


Рис. 2. Создание TextBox элемента.

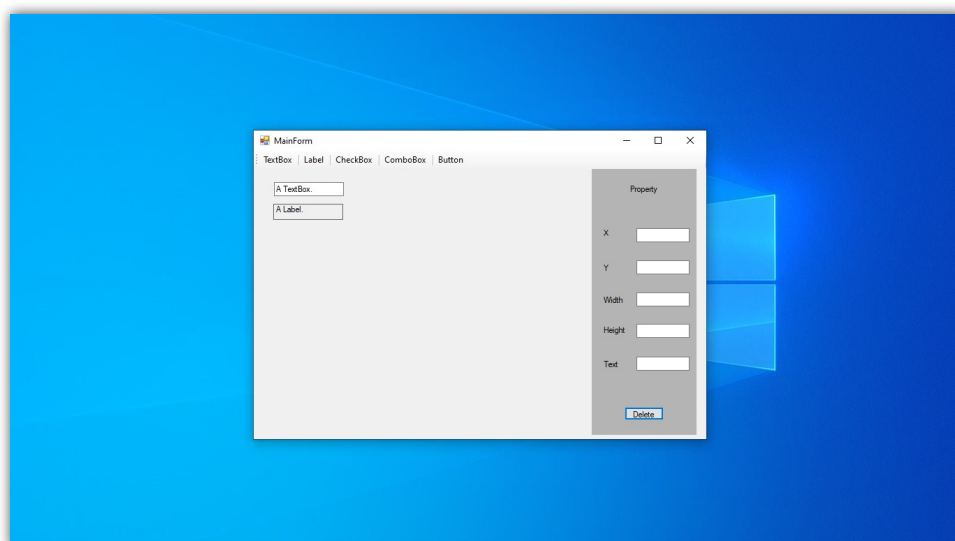


Рис. 3. Создание элемента Label.



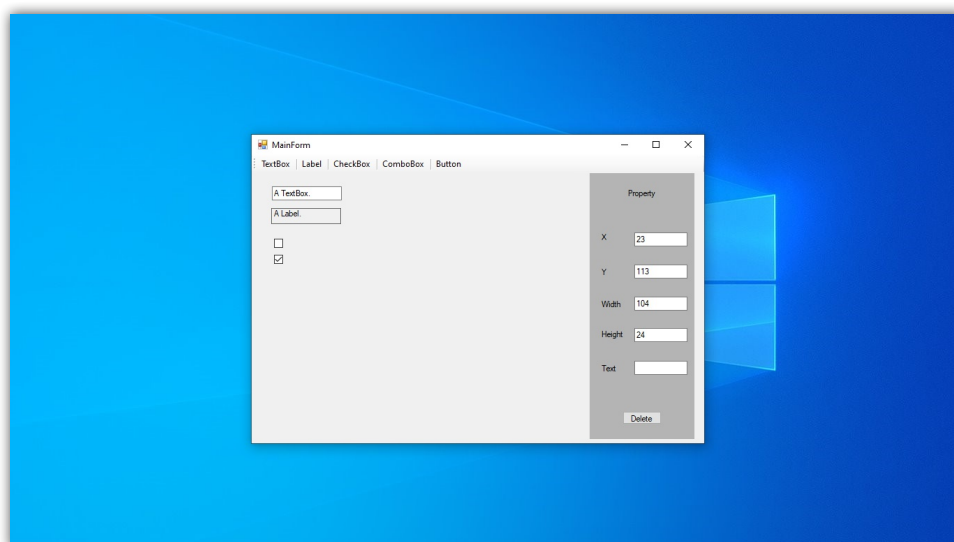


Рис. 4. Создание элементов CheckBox.

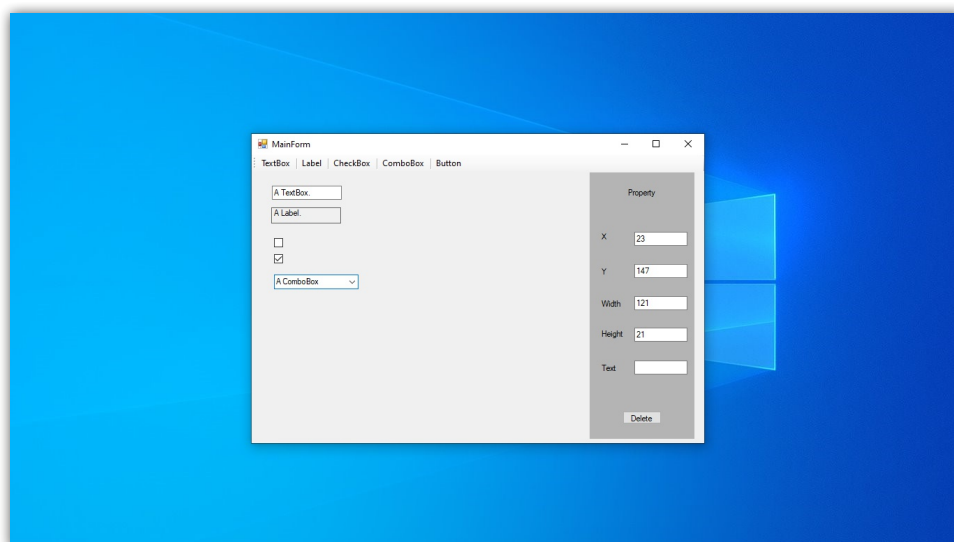


Рис. 5. Создание элемента ComboBox.

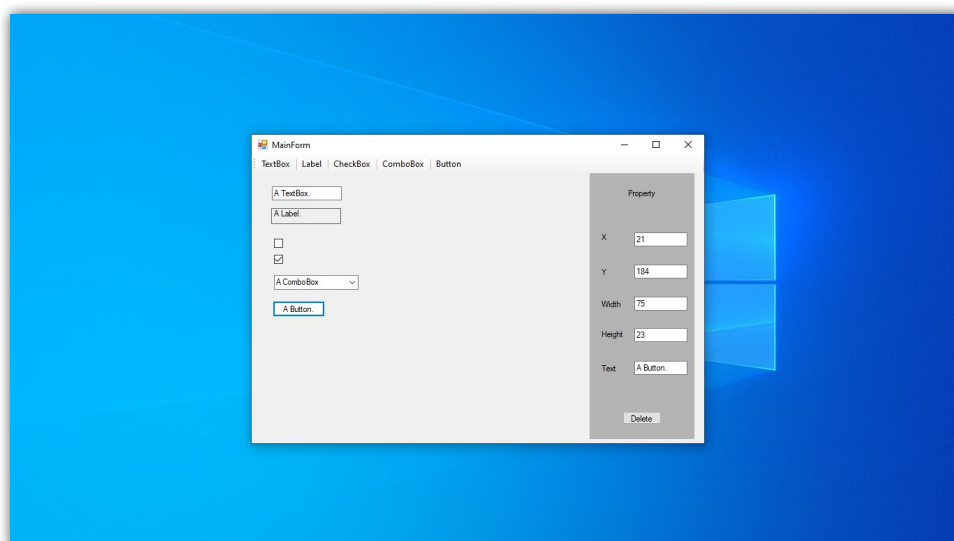


Рис. 6. Создание кнопки.