```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.IO;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
namespace WindowsFormsApplication2
{
        public partial class Form1 : Form
        {
        Button[,] buttons = new Button[3, 3];
        RichTextBox[,] rich = new RichTextBox[3, 3];
        Label[,] lab = new Label[3, 3];
        CheckBox[,] check = new CheckBox[3, 3];
        GroupBox[,] gb = new GroupBox[3, 3];//groupbox
        RadioButton[,] rb = new RadioButton[3, 3];
        public Form1()
        {
        InitializeComponent();
        }
        private void Form1_Load(object sender, EventArgs e)
        {
        }
        private void label1_MouseClick(object sender, MouseEventArgs e)
        {
        label1.Text = "X=" + e.X + ",Y= " + e.Y;
        }
        private void button1_Click(object sender, EventArgs e)
        {
        createbuttons();
        }
        private void createbuttons()
        {
        for (int i = 0; i < 3; i++)
        {
                for (int j = 0; j < 3; j++)
                {
                if (buttons[i, j] == null)
```

```
          {
          buttons[i, j] = new Button();
          buttons[i, j].Name = "button";
          buttons[i, j].Text = "Button";
          buttons[i, j].AutoSize = true;
          panel1.Controls.Add(buttons[i, j]);
          ControlExtension.Draggable(buttons[i, j], true);
          ColorDialog col = new ColorDialog();
          col.ShowDialog();
          buttons[i, j].BackColor = col.Color;
          i = i + 1;
          break;
          }
          }
          break;
}
}
private void button2_Click(object sender, EventArgs e)
{
createlabels();
}
private void createlabels()
{
for (int i = 0; i < 3; i++)
{
          for (int j = 0; j < 3; j++)
          {
          if (lab[i, j] == null)
          {
          lab[i, j] = new Label();
          lab[i, j].Name = "label";
          lab[i, j].Text = "Label";
          lab[i, j].AutoSize = true;
          panel1.Controls.Add(lab[i, j]);
          ControlExtension.Draggable(lab[i, j], true);
          ColorDialog col = new ColorDialog();
          col.ShowDialog();
          lab[i, j].BackColor = col.Color;
          i = i + 1;
          break;
          }
          }
          break;
```

```
}
}
private void button3_Click(object sender, EventArgs e)
{
createT();
}
public void createT()
{
for (int i = 0; i < 3; i++)
{
    for (int j = 0; j < 3; j++)
    {
    if (rich[i, j] == null)
    {
    rich[i, j] = new RichTextBox();
    rich[i, j].Name = "rich";
    rich[i, j].Text = "Textbox";
    rich[i, j].AutoSize = true;
    panel1.Controls.Add(rich[i, j]);
    ControlExtension.Draggable(rich[i, j], true);
    FontDialog fd = new FontDialog();
    if (fd.ShowDialog() == DialogResult.OK)
    {
        rich[i, j].Font = fd.Font;
    }
    ColorDialog col = new ColorDialog();
    col.ShowDialog();
    rich[i, j].BackColor = col.Color;
    i = i + 1;
    break;
    }
    }
    break;
}
}

private void button7_Click(object sender, EventArgs e)
{
createRadio();
}
private void createRadio()
{
for (int i = 0; i < 3; i++)
```

```
{
        for (int j = 0; j < 3; j++)
        {
        if (rb[i, j] == null)
        {
        rb[i, j] = new RadioButton();
        rb[i, j].Name = "rb";
        rb[i, j].Text = "radiobutton";
        rb[i, j].AutoSize = true;
        panel1.Controls.Add(rb[i, j]);
        ControlExtension.Draggable(rb[i, j], true);

        ColorDialog col = new ColorDialog();
        col.ShowDialog();
        rb[i, j].BackColor = col.Color;
        i = i + 1;
        break;
        }
        }
        break;
}
}
private void button6_Click(object sender, EventArgs e)
{
createGroup();
}

private void createGroup()
{
for (int i = 0; i < 3; i++)
{
        for (int j = 0; j < 3; j++)
        {
        if (gb[i, j] == null)
        {
        gb[i, j] = new GroupBox();
        gb[i, j].Name = "check";
        gb[i, j].Text = "Groupbox";
        gb[i, j].AutoSize = true;
        panel1.Controls.Add(gb[i, j]);
        ControlExtension.Draggable(gb[i, j], true);

        ColorDialog col = new ColorDialog();
```

```
    col.ShowDialog();
    gb[i, j].BackColor = col.Color;
    i = i + 1;
    break;
    }
    }
    break;
}
}
private void button5_Click(object sender, EventArgs e)
{
createCheckbox();
}
private void createCheckbox()
{
for (int i = 0; i < 3; i++)
{
    for (int j = 0; j < 3; j++)
    {
    if (check[i, j] == null)
    {
    check[i, j] = new CheckBox();
    check[i, j].Name = "check";
    check[i, j].Text = "Checkbox";
    check[i, j].AutoSize = true;
    panel1.Controls.Add(check[i, j]);
    ControlExtension.Draggable(check[i, j], true);

    ColorDialog col = new ColorDialog();
    col.ShowDialog();
    check[i, j].BackColor = col.Color;
    i = i + 1;
    break;
    }
    }
    break;
}
}

private void button4_Click(object sender, EventArgs e)
{
StreamWriter File = new StreamWriter("Xaml1.txt");
```

```
    File.Write("<?xml version= \"1.0\" encoding=\"utf - 8\" ?> \r\n  < ContentPage xmlns
=\"http://xamarin.com/schemas/2014/forms\" \r\n xmlns:x=\"http://schemas.
microsoft.com/winfx/2009/xaml\" \r\n xmlns:local=\"clr-namespace:App1\" \r\n x:Class =
\"App1.MainPage\"> \r\n <ContentPage.Content> \r\n <AbsoluteLayout
AbsoluteLayout.LayoutFlags = \"None\"> ");

    for (int i = 0; i < 3; i++)

    {

        for (int j = 0; j < 3; j++)

        {

        if (buttons[i, j] != null)

        {

        //coordinates

        File.Write("\t<Button");

        File.Write(" Text = \"");

        File.Write("Button");

        File.Write("\"");

        //layout bounds

        File.Write(" AbsoluteLayout.LayoutBounds = \"");

        //Top

        File.Write(buttons[i, j].Top.ToString());

        File.Write(",");

        //Left

        File.Write(buttons[i, j].Left.ToString());

        File.Write(",");

        //Height

        File.Write((buttons[i, j].Bottom - buttons[i, j].Top).ToString());
```

```
                File.Write(",");

                //Width

                File.Write((buttons[i, j].Right - buttons[i, j].Left).ToString());

                File.Write("\"");

                //color

                File.Write(" Color = \" #");

                File.Write(buttons[i, j].BackColor.R.ToString() + buttons[i,
        j].BackColor.G.ToString() + buttons[i, j].BackColor.B.ToString() + "\"" + ">\t");

                //Closing

                File.Write("</Button>");

                }

                if (lab[i, j] != null)

                {

                //coordinates

                File.Write("\t<Label");

                File.Write(" Text = \"");

                File.Write("Label");

                File.Write("\"");

                //layout bounds

                File.Write(" AbsoluteLayout.LayoutBounds = \"");

                //Top

                File.Write(lab[i, j].Top.ToString());

                File.Write(",");

                //Left

                File.Write(lab[i, j].Left.ToString());
```

```
File.Write(",");
//Height
File.Write((lab[i, j].Bottom - lab[i, j].Top).ToString());
File.Write(",");
//Width
File.Write((lab[i, j].Right - lab[i, j].Left).ToString());
File.Write("\"");
//color
File.Write(" Color = \" #");
File.Write(lab[i, j].BackColor.R.ToString() + lab[i, j].BackColor.G.ToString() + lab[i, j].BackColor.B.ToString() + "\"" + ">\t");
//Closing
File.Write("</Label>");
}
if (rich[i, j] != null)
{
//coordinates
File.Write("\t<Textbox");
File.Write(" Text = \"");
File.Write("textbox");
File.Write("\"");
//layout bounds
File.Write(" AbsoluteLayout.LayoutBounds = \"");
//Top
File.Write(rich[i, j].Top.ToString());
```

```
        File.Write(",");
        //Left
        File.Write(rich[i, j].Left.ToString());
        File.Write(",");
        //Height
        File.Write((rich[i, j].Bottom - rich[i, j].Top).ToString());
        File.Write(",");
        //Width
        File.Write((rich[i, j].Right - rich[i, j].Left).ToString());
        File.Write("\"");
        //color
        File.Write(" Color = \" #");
        File.Write(rich[i, j].BackColor.R.ToString() + rich[i, j].BackColor.G.ToString() +
rich[i, j].BackColor.B.ToString() + "\"" + ">\t");
        //Closing
        File.Write("</Button>");
        }
        if (check[i, j] != null)
        {
        //coordinates
        File.Write("\t<");
        File.Write("checkbox");
        File.Write("\"");
        //layout bounds
        File.Write(" AbsoluteLayout.LayoutBounds = \" Top = ");
File.Write(check[i, j].Top.ToString());

File.Write(",");

File.Write("Left = ");

File.Write(check[i, j].Left.ToString());

File.Write("Bottom = ");

File.Write(check[i, j].Bottom.ToString());

File.Write("Right = ");

File.Write(check[i, j].Right.ToString());

//width

File.Write("height of checkbox=");

File.Write(check[i, j].Bottom - check[i, j].Top);

File.Write("\t<");

//height

File.Write("width of checkbox=");

File.Write(check[i, j].Right - check[i, j].Left);

//color

File.Write("Color=");

File.Write(check[i, j].BackColor.ToString());

}

if (rich[i, j] != null)

{

//coordinates

File.Write("\t<");

File.Write("Textbox");

File.Write("\"");
```

```
//layout bounds

File.Write(" AbsoluteLayout.LayoutBounds = \" Top = ");

File.Write(rich[i, j].Top.ToString());

File.Write(",");

File.Write("Left = ");

File.Write(rich[i, j].Left.ToString());

File.Write("Bottom = ");

File.Write(rich[i, j].Bottom.ToString());

File.Write("Right = ");

File.Write(rich[i, j].Right.ToString());

//width

File.Write("height of textbox=");

File.Write(rich[i, j].Bottom - rich[i, j].Top);

File.Write("\t<");

//height

File.Write("width of textbox=");

File.Write(rich[i, j].Right - rich[i, j].Left);

//color

File.Write("Color=");

File.Write(rich[i, j].BackColor.ToString());

}

if (gb[i, j] != null)

{

//coordinates

File.Write("\t<");
File.Write("Groupbox");
File.Write("\"");
//layout bounds
File.Write(" AbsoluteLayout.LayoutBounds = \" Top = ");
File.Write(gb[i, j].Top.ToString());
File.Write(",");
File.Write("Left = ");
File.Write(gb[i, j].Left.ToString());
File.Write("Bottom = ");
File.Write(gb[i, j].Bottom.ToString());
File.Write("Right = ");
File.Write(gb[i, j].Right.ToString());
//width
File.Write("height of textbox=");
File.Write(gb[i, j].Bottom - gb[i, j].Top);
File.Write("\t<");
//height
File.Write("width of textbox=");
File.Write(gb[i, j].Right - gb[i, j].Left);
//color
File.Write("Color=");
File.Write(gb[i, j].BackColor.ToString());
}
if (rb[i, j] != null)
{
```

```
//coordinates
File.Write("\t<");
File.Write("RadioButton");
File.Write("\"");
//layout bounds
File.Write(" AbsoluteLayout.LayoutBounds = \" Top = ");
File.Write(rb[i, j].Top.ToString());
File.Write(",");
File.Write("Left = ");
File.Write(rb[i, j].Left.ToString());
File.Write("Bottom = ");
File.Write(rb[i, j].Bottom.ToString());
File.Write("Right = ");
File.Write(rb[i, j].Right.ToString());
//width
File.Write("height of textbox=");
File.Write(rb[i, j].Bottom - rb[i, j].Top);
File.Write("\t<");
//height
File.Write("width of textbox=");
File.Write(rb[i, j].Right - rb[i, j].Left);
//color
File.Write("Color=");
File.Write(rb[i, j].BackColor.ToString());
}
```

```
        }

    }

    File.Write("\t</ContentPage.Content> \r\n");

    File.Write("\t</ContentPage> \r\n");

    File.Write("\t</Absolute Layout> \r\n");

    File.Close();

    }

    private void panel1_Paint(object sender, PaintEventArgs e)

    {

    }


    }

}

*/


using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.IO;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;
```

```
namespace WindowsFormsApplication2

{

        public partial class Form1 : Form

        {

        public static String AppName = "App Name";

        ColorDialog col;

        FontDialog fd;

        int iSelButton = -1;

        int iSelLayout = -1;

        int iCtr = -1;

        public Form1()

        {

        InitializeComponent();

        }

        private void Form1_Load(object sender, EventArgs e)

        {

        col = new ColorDialog();

        fd = new FontDialog();

        label1.Text = AppName;

        Left = Top = 0;

        Width = Screen.PrimaryScreen.WorkingArea.Width;

        Height = Screen.PrimaryScreen.WorkingArea.Height;

        //prop_groupBox.Visible = false;

        //layoutlist.Visible = true;

        }
```

```
private void OnAddButton(object sender, EventArgs e)

{

if (-1 != iSelLayout)

{

        ButtonPanel();

        Button btn = new Button();

        btn.Name = "Button";

        btn.Text = "Click Me";

        btn.AutoSize = true;

        iCtr += 1;

        btn.Tag = iCtr;

        foreach (Control c in panel1.Controls)

        {

        if ((int)c.Tag == (int)iSelLayout)

        {

        c.Controls.Add(btn);

        }

        }

        ControlExtension.Draggable(btn, true);

        btn.Click += new EventHandler(this.DynamicButtonAction);/**/

}

}

private void ButtonPanel()

{
```

```
prop_groupBox.Text = "Button Properties";
elementlist.Visible = true;
text.Enabled = true;
background.Enabled = true;
keyboards.Enabled = false;
events.Enabled = false;
textBox1.Enabled = true;
label1.Text = "Caption";
}
void DynamicButtonAction(object sender, EventArgs e)
{
ButtonPanel();
Btn2Txt(sender);
}
void Btn2Txt(object sender)
{
Control c = sender as Control;
iSelButton = (int)c.Tag;
textBox1.Text = c.Text;
}
private void OnbColorClick(object sender, EventArgs e)
{
if (-1 != iSelButton)
{
        if (DialogResult.OK == col.ShowDialog())
```

```
    {
    foreach (Control c in panel1.Controls)
    {
    foreach (Control ctr in c.Controls)
    {
        if ((int)ctr.Tag == (int)iSelButton)
        {
       ctr.BackColor = col.Color;
        }
    }
    }
    }
}
}
private void button9_Click(object sender, EventArgs e)
{
if (-1 != iSelButton)
{
    if (fd.ShowDialog() == DialogResult.OK)
    {
    foreach (Control c in panel1.Controls)
    {
    foreach (Control ctr in c.Controls)
    {
        if ((int)ctr.Tag == (int)iSelButton)
```

```
               {
            ctr.Font = fd.Font;
               }
       }
       }
       }
}
}
private void button10_Click(object sender, EventArgs e)
{
if (-1 != iSelButton)
{
       if (DialogResult.OK == col.ShowDialog())
       {
       foreach (Control c in panel1.Controls)
       {
       foreach (Control ctr in c.Controls)
       {
               if ((int)ctr.Tag == (int)iSelButton)
               {
            ctr.ForeColor = col.Color;
               }
       }
       }
       }
```

```
}

}

private void textBox1_TextChanged(object sender, EventArgs e)

{

TextBox text1 = sender as TextBox;

foreach (Control c in panel1.Controls)

{

            foreach (Control ctr in c.Controls)

            {

            if ((int)ctr.Tag == (int)iSelButton)

            {

            ctr.Text = text1.Text;

            }

            }

}

}

private void ButtonWrite(StreamWriter File, Control c)

{

//coordinates

File.Write("\t<" + c.Name);

File.Write(" Text = \"");

File.Write(c.Text);

File.Write("\"");

//layout bounds

File.Write(" AbsoluteLayout.LayoutBounds = \"");
```

```
    //Top
    File.Write(c.Top.ToString());
    File.Write(",");
    //Left
    File.Write(c.Left.ToString());
    File.Write(",");
    //Height
    File.Write((c.Bottom - c.Top).ToString());
    File.Write(",");
    //Width
    File.Write((c.Right - c.Left).ToString());
    File.Write("\"");
    //color
    File.Write(" Color = \" #");
    File.Write(c.BackColor.R.ToString() + c.BackColor.G.ToString() +
c.BackColor.B.ToString() + "\"" + ">\t");
    //Closing
    File.Write("</" + c.Name + ">");
    }

    private void OnAddLabel(object sender, EventArgs e)
    {
    if (-1 != iSelLayout)
    {
        LabelPanel();
        Label lab = new Label();

        lab.Name = "Label";

        lab.Text = "Label";

        lab.AutoSize = true;

        iCtr += 1;

        lab.Tag = iCtr;

        foreach (Control c in panel1.Controls)

        {

        if ((int)c.Tag == (int)iSelLayout)

        {

        c.Controls.Add(lab);

        }

        }

        ControlExtension.Draggable(lab, true);

        lab.Click += new EventHandler(this.DynamicButtonAction);/**/

}

}

void DynamicLabelAction(object sender, EventArgs e)

{

LabelPanel();

Btn2Txt(sender);

}

private void LabelWrite(StreamWriter File, Control c)

{

//coordinates
```

```
        File.Write("\t<" + c.Name);
        File.Write(" Text = \"");
        File.Write(c.Text);
        File.Write("\"");
        //layout bounds
        File.Write(" AbsoluteLayout.LayoutBounds = \"");
        //Top
        File.Write(c.Top.ToString());
        File.Write(",");
        //Left
        File.Write(c.Left.ToString());
        File.Write(",");
        //Height
        File.Write((c.Bottom - c.Top).ToString());
        File.Write(",");
        //Width
        File.Write((c.Right - c.Left).ToString());
        File.Write("\"");
        //color
        File.Write(" Color = \" #");
        File.Write(c.BackColor.R.ToString() + c.BackColor.G.ToString() +
c.BackColor.B.ToString() + "\"" + ">\t");
        //Closing
        File.Write("</" + c.Name + ">");
        }
```

```
private void LabelPanel()

{

keyboards.Enabled = false;

events.Enabled = false;

prop_groupBox.Text = "Label Properties";

label1.Text = "Caption";

}

private void button3_Click_1(object sender, EventArgs e)

{

if (-1 != iSelLayout)

{

        TextPanel();

        TextBox text = new TextBox();

        text.Text = "Textbox";

        text.Name = "Entry";

        text.AutoSize = true;

        iCtr += 1;

        text.Tag = iCtr;

        foreach (Control c in panel1.Controls)

        {

        if ((int)c.Tag == (int)iSelLayout)

        {

        c.Controls.Add(text);

        }

        }
        ControlExtension.Draggable(text, true);

        text.Click += new EventHandler(this.DynamicButtonAction);

}

}

void DynamicTextAction(object sender, EventArgs e)

{

TextPanel();

Btn2Txt(sender);

}

private void TextBoxWrite(StreamWriter File, Control c)

{


//coordinates

File.Write("\t<" + c.Name);

File.Write(" Text = \"");

File.Write(c.Text);

File.Write("\"");

//layout bounds

File.Write(" AbsoluteLayout.LayoutBounds = \"");

//Top

File.Write(c.Top.ToString());

File.Write(",");

//Left

File.Write(c.Left.ToString());

File.Write(",");
```

```
    {
    //Height
    File.Write((c.Bottom - c.Top).ToString());
    File.Write(",");
    //Width
    File.Write((c.Right - c.Left).ToString());
    File.Write("\"");
    //color
    File.Write(" Color = \" #");
    File.Write(c.BackColor.R.ToString() + c.BackColor.G.ToString() +
c.BackColor.B.ToString() + "\"" + ">\t");
    //Closing
    File.Write("</" + c.Name + ">");
    }
    private void TextPanel()
    {
    prop_groupBox.Text = "Entry Properties";
    text.Enabled = true;
    events.Enabled = false;
    keyboards.Enabled = true;
    label1.Text = "PlaceHolder";
    }
    private void button7_Click(object sender, EventArgs e)
    {
    if (-1 != iSelLayout)
    {
```

```
    RadioPanel();

    RadioButton rb = new RadioButton();

    rb.Name = "Radio";

    rb.Text = "radiobutton";

    rb.AutoSize = true;

    iCtr += 1;

    rb.Tag = iCtr;

    foreach (Control c in panel1.Controls)

    {

    if ((int)c.Tag == (int)iSelLayout)

    {

    c.Controls.Add(rb);

    }

    }

    ControlExtension.Draggable(rb, true);

    rb.Click += new EventHandler(this.DynamicButtonAction);

}

}

void DynamicRadioAction(object sender, EventArgs e)

{

RadioPanel();

Btn2Txt(sender);

}

private void RadioPanel()

{
```

```
text.Enabled = true;
events.Enabled = false;
keyboards.Enabled = false;
prop_groupBox.Text = "Radio Properties";
label1.Text = "Caption";
}
private void RadioButtonWrite(StreamWriter File, Control c)
{
//coordinates
File.Write("\t<" + c.Name);
File.Write(" Text = \"");
File.Write(c.Text);
File.Write("\"");
//layout bounds
File.Write(" AbsoluteLayout.LayoutBounds = \"");
//Top
File.Write(c.Top.ToString());
File.Write(",");
//Left
File.Write(c.Left.ToString());
File.Write(",");
//Height
File.Write((c.Bottom - c.Top).ToString());
File.Write(",");
//Width
```

```
        File.Write((c.Right - c.Left).ToString());

        File.Write("\"");

        //color

        File.Write(" Color = \" #");

        File.Write(c.BackColor.R.ToString() + c.BackColor.G.ToString() +
c.BackColor.B.ToString() + "\"" + ">\t");

        //Closing

        File.Write("</" + c.Name + ">");

        }

        private void button6_Click(object sender, EventArgs e)

        {

        if (-1 != iSelLayout)

        {

                GroupPanel();

                GroupBox gb = new GroupBox();

                gb.Name = "Group";

                gb.Text = "Groupbox";

                gb.AutoSize = true;

                iCtr += 1;

                gb.Tag = iCtr;

                foreach (Control c in panel1.Controls)

                {

                if ((int)c.Tag == (int)iSelLayout)

                {

                c.Controls.Add(gb);
        }
        }
        ControlExtension.Draggable(gb, true);
        gb.Click += new EventHandler(this.DynamicGroupAction);
}
}
void DynamicGroupAction(object sender, EventArgs e)
{
GroupPanel();
Btn2Txt(sender);
}
private void GroupBoxWrite(StreamWriter File, Control c)
{
//coordinates
File.Write("\t<" + c.Name);
File.Write(" Text = \"");
File.Write(c.Text);
File.Write("\"");
//layout bounds
File.Write(" AbsoluteLayout.LayoutBounds = \"");
//Top
File.Write(c.Top.ToString());
File.Write(",");
//Left
File.Write(c.Left.ToString());
```

```
    File.Write(",");
    //Height
    File.Write((c.Bottom - c.Top).ToString());
    File.Write(",");
    //Width
    File.Write((c.Right - c.Left).ToString());
    File.Write("\"");
    //color
    File.Write(" Color = \" #");
    File.Write(c.BackColor.R.ToString() + c.BackColor.G.ToString() +
c.BackColor.B.ToString() + "\"" + ">\t");
    //Closing
    File.Write("</" + c.Name + ">");
    }
    private void GroupPanel()
    {
    text.Enabled = true;
    events.Enabled = false;
    keyboards.Enabled = false;
    prop_groupBox.Text = "Group Properties";
    label1.Text = "Caption";
    }
    private void button5_Click(object sender, EventArgs e)
    {
    if (-1 != iSelLayout)
```

```
{
        CheckPanel();
        CheckBox check = new CheckBox();
        check.Name = "Check";
        check.Text = "Checkbox";
        check.AutoSize = true;
        iCtr += 1;
        check.Tag = iCtr;
        foreach (Control c in panel1.Controls)
        {
        if ((int)c.Tag == (int)iSelLayout)
        {
        c.Controls.Add(check);
        }
        }
        ControlExtension.Draggable(check, true);
        check.Click += new EventHandler(this.DynamicCheckAction);
}
}
private void CheckPanel()
{
prop_groupBox.Text = "Check Box Properties";
text.Enabled = true;
events.Enabled = false;
keyboards.Enabled = false;
```

```
label1.Text = "Caption";
}
void DynamicCheckAction(object sender, EventArgs e)
{
CheckPanel();
Btn2Txt(sender);
}
private void CheckboxWrite(StreamWriter File, Control c)
{
//coordinates
File.Write("\t<" + c.Name);
File.Write(" Text = \"");
File.Write(c.Text);
File.Write("\"");
//layout bounds
File.Write(" AbsoluteLayout.LayoutBounds = \"");
//Top
File.Write(c.Top.ToString());
File.Write(",");
//Left
File.Write(c.Left.ToString());
File.Write(",");
//Height
File.Write((c.Bottom - c.Top).ToString());
File.Write(",");
```

```
    //Width
    File.Write((c.Right - c.Left).ToString());
    File.Write("\"");
    //color
    File.Write(" Color = \" #");
    File.Write(c.BackColor.R.ToString() + c.BackColor.G.ToString() + c.BackColor.B.ToString() + "\"" + ">\t");
    //Closing
    File.Write("</" + c.Name + ">");
    }
    private void button4_Click(object sender, EventArgs e)
    {
    MessageBox.Show("Main.Write");
    StreamWriter File = new StreamWriter("Xaml1.txt");
    File.Write("<?xml version= \"1.0\" encoding=\"utf - 8\" ?> \r\n  < ContentPage xmlns =\"http://xamarin.com/schemas/2014/forms\" \r\n xmlns:x=\"http://schemas.microsoft.com/winfx/2009/xaml\" \r\n xmlns:local=\"clr-namespace:App1\" \r\n x:Class = \"App1.MainPage\"> \r\n <ContentPage.Content> \r\n ");

    foreach (Control c in panel1.Controls)
    {
        if (c.Name == "AbsLayout")
        {
        AbsLayoutWrite(File, c);
        }
```

```
            /*/*else if (c.Name == "Label")
            {
            LabelWrite(File, c);
            }
            else if ( (c.Name == "Editor") || (c.Name == "Entry"))
            {
            TextBoxWrite(File, c);
            }
            else if (c.Name == "Radio")
            {
            RadioButtonWrite(File, c);
            }
            else if (c.Name == "Check")
            {
            CheckboxWrite(File, c);
            }
            else if (c.Name == "Group")
            {
            GroupBoxWrite(File, c);
            }*/
}


File.Write("\t</ContentPage.Content> \r\n");
File.Write("\t</ContentPage> \r\n");
File.Close();

}

private void OnLayouts(object sender, EventArgs e)

{

elementlist.Visible = false;

layoutlist.Visible = true;

prop_groupBox.Text = "Layout Properties";

text.Visible = false;

background.Visible = false;

events.Visible = false;

keyboards.Visible = false;

}

private void OnElements(object sender, EventArgs e)

{

elementlist.Visible = true;

layoutlist.Visible = false;

prop_groupBox.Text = "Elements Properties";

text.Visible = true;

background.Visible = true;

events.Visible = true;

keyboards.Visible = true;

}

private void BtnAbsLayout_Click(object sender, EventArgs e)

{

Panel AbsLayout = new Panel();
```

```
AbsLayoutPanel();

AbsLayout.Top = panel1.Top;
AbsLayout.Left = 100 + (panel1.Left / 2);
AbsLayout.Height = panel1.Height / 2;
AbsLayout.Width = panel1.Width;
AbsLayout.BackColor = Color.AntiqueWhite;
AbsLayout.Name = "AbsLayout";
iCtr += 1;
AbsLayout.Tag = iCtr;
panel1.Controls.Add(AbsLayout);
ControlExtension.Draggable(AbsLayout, true);
AbsLayout.Click += new EventHandler(this.DynamicAbsLayoutAction);
}
private void AbsLayoutPanel()
{
prop_groupBox.Text = "Absolute Layout Properties";
text.Enabled = true;
events.Enabled = false;
keyboards.Enabled = false;
textBox1.Enabled = false;
}
private void DynamicAbsLayoutAction(object sender, EventArgs e)
{
AbsLayoutPanel();
```

```
    }

LayoutSelector(sender);
}
void AbsLayoutWrite(StreamWriter File, Control ctr)
{

MessageBox.Show("Abslayour.Write");
File.Write("\r\n <" + ctr.Name + " " + ctr.Name + ".LayoutFlags = \"None\"> ");
foreach (Control c in ctr.Controls)
{
    if (c.Name == "Button")
    {
    ButtonWrite(File, c);
    }
    else if (c.Name == "Label")
    {
    LabelWrite(File, c);
    }
    else if ((c.Name == "Editor") || (c.Name == "Entry"))
    {
    TextBoxWrite(File, c);
    }
    else if (c.Name == "Radio")
    {
    RadioButtonWrite(File, c);
    }
```

```
        else if (c.Name == "Check")
        {
        CheckboxWrite(File, c);
        }
        else if (c.Name == "Group")
        {
        GroupBoxWrite(File, c);
        }
    }
    File.Write("\t</" + ctr.Name + "> \r\n");
    }
    private void LayoutSelector(object sender)
    {
    Control c = sender as Control;
    iSelLayout = (int)c.Tag;
    }
    private void label4_Click(object sender, EventArgs e)
    {
    /*var form = new Form2();
    form.Show();*/
    }
    }
}
```