

Course 2DCis: 2D-Computer Graphics with C#

Chapter C5: The Complete Code of the Controls Project

Copyright © by V. Miszalok, last update: 11-12-2007

Copy all this code into an empty Form1.cs of a new Windows Application C#-project controls1 and clear Form1.Designer.cs and Program.cs.

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

public class Form1 : Form
{
    [STAThread] static void Main() { Application.Run( new Form1() ); }
    const Int32 nButtons = 3, nTrackBars = 3, nCheckBoxes = 3, nRadioButtons = 3;
    Button [] button = new Button[nButtons];
    TrackBar[] trackbar = new TrackBar[nTrackBars];
    Label [] label = new Label[nTrackBars];
    CheckBox[] checkbox = new CheckBox[nCheckBoxes];
    static RadioButton[] radiobutton = new RadioButton[nRadioButtons];
    static Color linecolor = Color.Black;
    static Pen pen = new Pen( linecolor, 1 );
    static Panel panel = new Panel();
    static Graphics g;
    static Random r = new Random();
    Timer myTimer = new Timer();

    public Form1()
    {
        BackColor = Color.White;
        Text = "GUI";
        Int32 i;
        for ( i=0; i < nButtons; i++ )
        {
            button[i] = new Button(); Controls.Add( button[i] );
            button[i].Click += new EventHandler( button_handler );
        }
        button[0].Text = "Start";
        button[1].Text = "Stop";
        button[2].Text = "Clear";
        for ( i=0; i < nTrackBars; i++ )
        {
            trackbar[i] = new TrackBar(); Controls.Add( trackbar[i] );
            label [i] = new Label(); Controls.Add( label[i] );
            trackbar[i].AutoSize = false;
            trackbar[i].TickStyle = TickStyle.None;
            trackbar[i].ValueChanged += new EventHandler( trackbar_handler );
            label [i].TextAlign = ContentAlignment.TopCenter;
        }
        trackbar[0].Name = label[0].Text = "TimerInterval";
        trackbar[1].Name = label[1].Text = "PenThickness";
        trackbar[2].Name = label[2].Text = "Brightness";
        trackbar[0].Minimum = 13; trackbar[0].Maximum = 1000;
        trackbar[1].Minimum = 1; trackbar[1].Maximum = 20;
        trackbar[2].Minimum = 0; trackbar[2].Maximum = 255;
        for ( i=0; i < nCheckBoxes; i++ )
        {
            checkbox[i] = new CheckBox(); Controls.Add( checkbox[i] );
            checkbox[i].TextAlign = ContentAlignment.MiddleCenter;
            checkbox[i].Click += new EventHandler( checkbox_handler );
        }
        checkbox[0].Text = "Red";
        checkbox[1].Text = "Green";
        checkbox[2].Text = "Blue";
        for ( i=0; i < nRadioButtons; i++ )
        {
            radiobutton[i] = new RadioButton(); Controls.Add( radiobutton[i] );
            radiobutton[i].TextAlign = ContentAlignment.MiddleCenter;
        }
        radiobutton[0].Text = "Lines"; radiobutton[0].Checked = true;
        radiobutton[1].Text = "Rectangles";
        radiobutton[2].Text = "Ellipses";

        foreach ( Control c in Controls )
        {
            c.BackColor = Color.Gray;
            if ( c.Text == "" ) c.Text = "nothing";
        }
        Controls.Add( panel ); //Put a drawing space on Form1
        myTimer.Tick += new EventHandler( OnTimer );
        myTimer.Interval = 1;
        Width = 800; Height = 600;
    }
}
```

```

protected override void OnResize( EventArgs e )
{
    Int32 w = ClientRectangle.Width / 5;
    Int32 h = ClientRectangle.Height / (Controls.Count-1);
    Int32 i, top = 1;
    for ( i=0; i < Controls.Count-1; i++ )
    {
        Controls[i].Top    = top;
        Controls[i].Left   = 2;
        Controls[i].Width  = w;
        Controls[i].Height = h - 2;
        top += h;
    }
    for ( i=0; i < nTrackBars; i++ ) trackbar[i].Height = h;
    panel.Location = new Point( w+2, 0 );
    panel.Size = new Size( ClientRectangle.Width-panel.Location.X, ClientRectangle.Height );
    g = panel.CreateGraphics();
    g.Clear( SystemColors.Window );
}

protected void button_handler( object sender, System.EventArgs e )
{
    switch( ((Button)sender).Text )
    {
        case "Start": myTimer.Start(); break;
        case "Stop" : myTimer.Stop(); break;
        case "Clear": g.Clear( SystemColors.Window ); break;
    }
}

protected void trackbar_handler( object sender, System.EventArgs e )
{
    Int32 value = ((TrackBar)sender).Value;
    switch( ((TrackBar)sender).Name )
    {
        case "TimerInterval":
            myTimer.Interval = value;
            label[0].Text = "TimerInterval = " + value.ToString(); break;
        case "PenThickness":
            pen.Width = value );
            label[1].Text = "PenThickness = " + value.ToString(); break;
        case "Brightness":
            label[2].Text = "Brightness = " + value.ToString();
            checkbox_handler( sender, e ); break; //call checkbox_handler
    }
}

protected void checkbox_handler( object sender, System.EventArgs e )
{
    Int32 v;
    if ( sender == trackbar[2] ) v = trackbar[2].Value; //call from "Brightness"
    else v = trackbar[2].Value = 255; //call from CheckBox
    linecolor = Color.Black; //start color
    if ( checkbox[0].Checked ) linecolor = Color.FromArgb( v,0,0 );
    if ( checkbox[1].Checked ) linecolor = Color.FromArgb( linecolor.R,v,0 );
    if ( checkbox[2].Checked ) linecolor = Color.FromArgb( linecolor.R,linecolor.G,v );
    //If all checkboxes are empty, the linecolor remains black. Replace it by gray.
    if ( linecolor == Color.Black && v > 0 ) linecolor = Color.FromArgb( v,v,v );
    pen.Color = linecolor;
}

protected static void OnTimer( Object myObject, EventArgs myEventArgs )
{
    Int32 w = r.Next( panel.Width/2 ), h = r.Next( panel.Height/2 );
    Int32 x = r.Next( panel.Width-w ), y = r.Next( panel.Height-h );
    if ( radiobutton[0].Checked ) g.DrawLine ( pen, x, y, x+w, y+h );
    else if ( radiobutton[1].Checked ) g.DrawRectangle( pen, x, y, w, h );
    else if ( radiobutton[2].Checked ) g.DrawEllipse ( pen, x, y, w, h );
}
}

```