

# Double buffer draw



```
using System;
using System.Drawing;
using System.Collections;
using System.ComponentModel;
using System.Windows.Forms;
using System.Data;

public class Form1 : System.Windows.Forms.Form {
private System.ComponentModel.Container components = null;

public Form1() {
InitializeComponent();
}

protected override void Dispose( bool disposing ) {
if( disposing ) {
if (components != null) {
components.Dispose();
}
}
base.Dispose( disposing );
}

private void InitializeComponent() {
this.components = new System.ComponentModel.Container();
this.Size = new System.Drawing.Size(300,300);
this.Text = "Form1";
}

static void Main() {
Application.Run(new Form1());
}

protected override void OnPaint(PaintEventArgs e) {
```

```
Graphics displayGraphics = e.Graphics;
Random r = new Random();
Image i = new Bitmap(ClientRectangle.Width,
ClientRectangle.Height);
Graphics g = Graphics.FromImage(i);

g.FillRectangle(Brushes.White, ClientRectangle);

for (int x = 0; x < ClientRectangle.Width; x++) { for (int y =
0; y < ClientRectangle.Height; y += 10) { Color c =
Color.FromArgb (r.Next(25), r.Next(55), r.Next(5)); Pen p =
new Pen(c, 1); g.DrawLine(p, new
Point(ClientRectangle.Width/2, ClientRectangle.Height/2), new
Point(x, y)); p.Dispose(); } } displayGraphics.DrawImage(i,
ClientRectangle); i.Dispose(); } } [/csharp]
```