

Clip Text

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Drawing.Drawing2D;
using System.Text;
using System.Windows.Forms;

class FontMenuForm : Form {
protected string strText = "Clip Text";
protected Font font = new Font("Times New Roman", 24);

public static void Main() {
Application.Run(new FontMenuForm());
}
public FontMenuForm() {
ResizeRedraw = true;
}
protected void DoPage(Graphics grfx, Color clr, int cx, int
cy) {
GraphicsPath path = new GraphicsPath();
float fFontSize = PointsToPageUnits(grfx, font);

path.AddString(strText, font.FontFamily, (int)font.Style,
fFontSize, new PointF(0, 0), new StringFormat());

grfx.SetClip(path);

RectangleF rectfBounds = path.GetBounds();
grfx.TranslateClip(
(cx - rectfBounds.Width) / 2 - rectfBounds.Left,
(cy - rectfBounds.Height) / 2 - rectfBounds.Top);

Random rand = new Random();
```

```
for (int y = 0; y < cy; y++) { Pen pen = new
Pen(Color.FromArgb(rand.Next(255), rand.Next(255),
rand.Next(255))); grfx.DrawBezier(pen, new Point(0, y), new
Point(cx / 3, y + cy / 3), new Point(2 * cx / 3, y - cy / 3),
new Point(cx, y)); } } public float GetAscent(Graphics grfx,
Font font) { return font.GetHeight(grfx) *
font.FontFamily.GetCellAscent(font.Style) /
font.FontFamily.GetLineSpacing(font.Style); } public float
GetDescent(Graphics grfx, Font font) { return
font.GetHeight(grfx) *
font.FontFamily.GetCellDescent(font.Style) /
font.FontFamily.GetLineSpacing(font.Style); } public float
PointsToPageUnits(Graphics grfx, Font font) { float fFontSize;
if (grfx.PageUnit == GraphicsUnit.Display) fFontSize = 100 *
font.SizeInPoints / 72; else fFontSize = grfx.DpiX *
font.SizeInPoints / 72; return fFontSize; } } [/csharp]
```