

Widen a Path

```
using System;
using System.Drawing;
using System.Drawing.Drawing2D;
using System.Windows.Forms;

class WidenPath: Form
{
    GraphicsPath path= new GraphicsPath();

    public static void Main()
    {
        Application.Run(new WidenPath());
    }
    public WidenPath()
    {
        ResizeRedraw = true;

        path.AddLines(new Point[] { new Point(20, 10),
            new Point(50, 50),
            new Point(80, 10) });

        path.AddPolygon(new Point[] { new Point(20, 30),
            new Point(50, 70),
            new Point(80, 30) });
    }
    protected override void OnPaint(PaintEventArgs pea)
    {
        DoPage(pea.Graphics,          ForeColor, ClientSize.Width,
            ClientSize.Height);
    }
    protected void DoPage(Graphics grfx, Color clr, int cx, int
    cy)
    {
        grfx.ScaleTransform(cx / 300f, cy / 200f);
    }
}
```

```
for (int i = 0; i < 6; i++) { GraphicsPath pathClone =
(GraphicsPath) path.Clone(); Matrix matrix = new Matrix(); Pen
penThin = new Pen(clr, 1); Pen penThick = new Pen(clr, 5); Pen
penWiden = new Pen(clr, 7.5f); Brush brush = new
SolidBrush(clr); matrix.Translate((i % 3) * 100, (i / 3) *
100); if (i < 3) pathClone.Transform(matrix); else
pathClone.Widen(penWiden, matrix); switch (i % 3) { case 0:
grfx.DrawPath(penThin, pathClone); break; case 1:
grfx.DrawPath(penThick, pathClone); break; case 2:
grfx.FillPath(brush, pathClone); break; } } } } [/csharp]
```