

Paint Owner-Draw Buttons

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

class OwnerDrawButtons: Form
{
    readonly int cxImage, cyImage;
    readonly int cxBtn, cyBtn, dxBtn;
    readonly Button btnLarger, btnSmaller;

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

    public OwnerDrawButtons()
    {
        ResizeRedraw = true;

        cxImage = 4 * Font.Height;
        cyImage = 4 * Font.Height;
        cxBtn = cxImage + 8;
        cyBtn = cyImage + 8;
        dxBtn = Font.Height;

        btnLarger = new Button();
        btnLarger.Parent = this;
        btnLarger.Size = new Size(cxBtn, cyBtn);
        btnLarger.Click += new EventHandler(ButtonLargerOnClick);
        btnLarger.Paint += new PaintEventHandler(ButtonOnPaint);

        btnSmaller = new Button();
        btnSmaller.Parent = this;
        btnSmaller.Size = new Size(cxBtn, cyBtn);
        btnSmaller.Click += new EventHandler(ButtonSmallerOnClick);
```

```

btnSmaller.Paint += new PaintEventHandler(ButtonOnPaint);

OnResize(EventArgs.Empty);
}
protected override void OnResize(EventArgs ea)
{
base.OnResize(ea);

btnLarger.Location =
new Point(ClientSize.Width / 2 - cxBtn - dxBtn / 2,
(ClientSize.Height - cyBtn) / 2);
btnSmaller.Location =
new Point(ClientSize.Width / 2 + dxBtn / 2,
(ClientSize.Height - cyBtn) / 2);
}
void ButtonLargerOnClick(object obj, EventArgs ea)
{
Console.WriteLine("clicked large");
}
void ButtonSmallerOnClick(object obj, EventArgs ea)
{
Console.WriteLine("clicked small");
}
void ButtonOnPaint(object obj, PaintEventArgs pea)
{
Button btn = (Button) obj;
Graphics grfx = pea.Graphics;

ControlPaint.DrawButton(grfx, 0, 0, cxBtn, cyBtn,
(btn == (Button) GetChildAtPoint(
PointToClient(
MousePosition))) &&
btn.Capture ? ButtonState.Pushed : ButtonState.Normal);

GraphicsState grfxstate = grfx.Save();

grfx.TranslateTransform((cxBtn - cxImage) / 2, (cyBtn -
cyImage) / 2);

```

```
DrawLargerButton(grfx, cxImage, cyImage);

grfx.Restore(grfxstate);

if (btn.Focused)
ControlPaint.DrawFocusRectangle(grfx,
new Rectangle((cxBtn - cxImage) / 2 + cxImage / 16,
(cyBtn - cyImage) / 2 + cyImage / 16,
7 * cxImage / 8, 7 * cyImage / 8));
}
void DrawLargerButton(Graphics grfx, int cx, int cy)
{
Brush brush = new SolidBrush(btnLarger.ForeColor);
Pen pen = new Pen(btnLarger.ForeColor);

grfx.TranslateTransform(cx / 2, cy / 2);

for (int i = 0; i < 4; i++) { grfx.DrawLine(pen, 0, 0, cx / 4,
0); grfx.FillPolygon(brush, new Point[] { new Point(cx / 4, -
cy / 8), new Point(cx / 2, 0), new Point(cx / 4, cy / 8)});
grfx.RotateTransform(90); } } } [/csharp]
```