in this case create only two instances. The complete folder class shown next creates a folder instance with one background color or the other and has a public *draw* method that draws the folder at the point we specify.

```
public class Folder extends JPanel {
    private Color color;
    final int W = 50, H = 30;
    final int tableft = 0, tabheight=4, tabwidth=20,
              tabslant=3;
    public Folder(Color c) {
        color = c;
    }
//--------------
    public void draw(Graphics g, int tx, int ty,
                     String name) {
        g.setColor(Color.black);              //outline
        g.drawRect(tx, ty, W, H);
        g.drawString(name, tx, ty + H+15);    //title
        g.setColor(Color.white);
        g.drawLine (tx, ty, tx+W, ty);

        Polygon poly = new Polygon();
        poly.addPoint (tx+tableft,ty);
        //etc...
        g.drawPolygon (poly);

        g.setColor(color);                    //fill rectangle
        g.fillRect(tx+1, ty+1, W-1, H-1);
        g.fillPolygon (poly);
        g.setColor(Color.white);
        g.drawLine (tx, ty, tx+W, ty);

        g.setColor(Color.black);              //shadow lines
        g.drawLine(tx, ty+H+1, tx+W-1, ty+H+1);
        g.drawLine(tx+W+1, ty, tx+W+1, ty+H);

        g.setColor(Color.white);              //highlight lines
        g.drawLine(tx+1, ty+1, tx+W-1, ty+1);
        g.drawLine(tx+1, ty+1, ty+H-1);
    }
}
```

To use a Flyweight class like this, our calling program must calculate the position of each folder as part of its paint routine and then pass the coordinates to the folder instance. This is actually rather common, since we need a different layout depending on the window's dimensions and we do not want to have to keep telling each instance where its new location is going to be. Instead, we compute it dynamically during the paint routine.