

COMPOUNDING SHAPES IN JAVA

The `acm.graphics` package contains many classes for drawing different shapes: `GArc`, `GLine`, `GOval`, and `GRect` are the four most basic choices. If you look at the full documentation, you'll see several other much less useful choices, of which some are not even shapes.

One particularly useful class is `GCompound`. While not itself a shape, it allows you to combine several shapes into a single object, which can then be manipulated as a group.

```
GCompound()
```

(Constructor) Constructs a compound object, initially containing no shapes.

```
void add(GObject shape)
```

Adds `shape` as one of the objects that is part of this compound object.

```
void move(double dx, double dy)
```

Moves all shapes in this compound object `dx` pixels to the right and `dy` pixels down. A negative value for a parameter moves all shapes left/up.

[Figure 6.4](#) contains a program where a `GCompound` is useful. In this program, we combine a circle, a rectangle, and two lines into one `GCompound` that has the appearance of a hot air balloon. We then animate the balloon's descent down to the bottom of the window, where it lands (i.e., stops). (Note the usage of `getHeight` at the end of the `while` loop's condition: Because we don't specify the object to which to apply `getHeight`, the compiler assumes we want to apply it to `this` — i.e., the window. Luckily, the window has such a method.)

Figure 6.4: The `MovingBalloon` program.

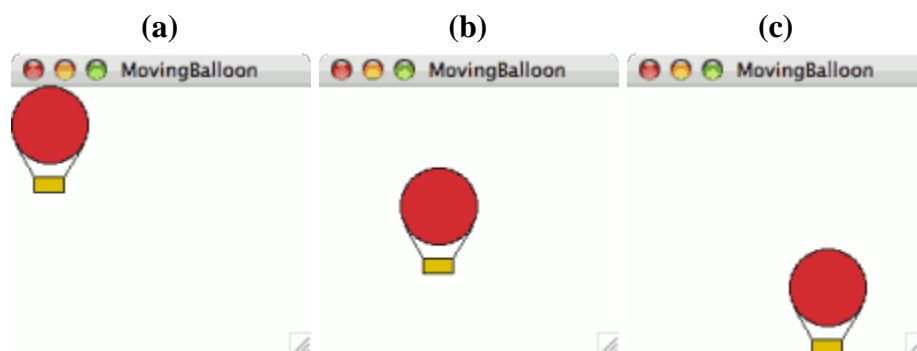
```
1 import acm.program.*;
2 import acm.graphics.*;
3 import java.awt.*;
4
5 public class MovingBalloon extends GraphicsProgram {
6     public void run() {
7         // Create the circle that will appear as the balloon's top
8         GOval ball = new GOval(0, 0, 50, 50);
9         ball.setFilled(true);
10        ball.setFill(new Color(208, 48, 48));
```

```

11
12     // Create a rectangle that will appear as the balloon's basket
13     GRect basket = new GRect(15, 60, 20, 10);
14     basket.setFilled(true);
15     basket.setFill-color(new Color(224, 192, 0));
16
17     // Now put these together, along with ropes connecting them
18     GCompound balloon = new GCompound();
19     balloon.add(ball);
20     balloon.add(new GLine(3, 39, 15, 60)); // left rope
21     balloon.add(new GLine(48, 39, 35, 60)); // right rope
22     balloon.add(basket);
23     add(balloon);
24
25     // Move the whole combination down and right until it lands on
26     // the window's bottom. (We add 70 to balloon's y because
27     // bottom of basket is 70 pixels below balloon's origin.)
28     while(balloon.getY() + 70 < getHeight()) {
29         pause(40);
30         balloon.move(1, 1);
31     }
32 }
33 }

```

Figure 6.5: Running MovingBalloon.



Source : <http://www.toves.org/books/java/ch06-graphics/index.html>