

```
1  /*****
2  *   Actor
3  *****/
4  public class Actor {
5      protected PImage[] imgs;
6      protected int x, y;
7      protected int w, h;
8      protected int index;
9      protected float scale=1.0;
10     protected Terminal<Boolean> visible=new Terminal<Boolean>(true);
11
12     public Actor(JSONObject src) {
13
14         if (!src.isNull("images")) {
15             JSONArray jImages=src.getJSONArray("images");
16             imgs=new PImage[jImages.size()];
17             for (int i=0; i<jImages.size(); i++) {
18                 imgs[i]=loadImage("images\\"+jImages.getString(i));
19             }
20             w=(imgs[0].width);
21             h=imgs[0].height;
22             visible.write(true);
23         } else
24             visible.write(false);
```

C:/nqCbtp/JavaApplication8/src/Terminal.java

```
25     x=src.getInt("x");
26     y=src.getInt("y");
27     index=0;
28     if (!src.isNull("visible"))
29         visible.write(src.getBoolean("visible"));
30     if (!src.isNull("scale"))
31         scale=src.getFloat("scale");
32     w=Math.round((float)w*scale);
33     h=Math.round((float)h*scale);
34 }
35
36 public void draw() {
37     if (visible.read()) {
38         pushMatrix();
39         scale(scale);
40         image(imgs[index], x/scale, y/scale);
41         popMatrix();
42     }
43 }
44
45 public boolean isX(int X, int x1, int x2) {
46     return ( (x+x1)<X) && ((x+x2)>X) && visible.read());
47 }
48
```

C:/nqCbtp/JavaApplication8/src/Terminal.java

```
49  public boolean isY(int Y, int y1, int y2) {
50      return ( ((y+y1)<Y) && ((y+y2)>Y) && visible.read());
51  }
52
53  public Terminal getTerminal(String terminalName) {
54      if (terminalName.equals("visible"))
55          return visible;
56      else
57          return null;
58  }
59 }
```