```
 1
 2 public class Tank extends Actor implements Runnable {   3
 4      protected Terminal<Integer> level = new Terminal<Integer>(0);
 5    private String inFlowTargetName = "";
 6     private String outFlowTargetName = "";
 7     private int minLevel = 0;
 8     private int maxLevel = 100;
 9     private int surface = 1;
10
11     public Tank(JSONObject src) {
12          super(src);
13         if (!src.isNull("inFlow"))
14             inFlowTargetName = src.getString("inFlow");
15       if (!src.isNull("outFlow"))
16            outFlowTargetName = src.getString("outFlow");
17       if (!src.isNull("minLevel"))
18            minLevel = src.getInt("minLevel");
19        if (!src.isNull("maxLevel"))
20            maxLevel = src.getInt("maxLevel");
21        if (!src.isNull("surface")) {
22           surface = src.getInt("surface");
23        }
24     }
```

```
25
26     void run() {
27          int totalFlow;
28          int quantity = 0;
29          int maxQuantity = 100 * maxLevel * surface;
30          Terminal<Integer> inValve = null;
31          Terminal<Integer> outValve = null;
32          if (!inFlowTargetName.equals("")) {
33              inValve =
actorsContainer.get(actorsIndex.get(inFlowTargetName)).getTerminal("flow");
34          }
35          if (!outFlowTargetName.equals("")) {
36              outValve =
actorsContainer.get(actorsIndex.get(outFlowTargetName)).getTerminal("flow");
37          }
38
39          for (;;) {
40              totalFlow = ((inValve != null) ? inValve.read() : 0) - ((outValve != null)
? outValve.read() : 0);
41              quantity += totalFlow;
42              if (quantity < minLevel) {
43                  quantity = minLevel;
44              }
45              if (quantity > maxQuantity) {
```

```
46                    quantity = maxQuantity;
47                }
48            level.write(quantity / surface / maxLevel);
49            try {
50                java.lang.Thread.sleep(samplingTime);
51            } catch (Exception e) {
52                println(e);
53            }
54        }
55    }
56
57    public Terminal getTerminal(String terminalName) {
58        if
59      (terminalName.equals("level")) {
60            return level;
61        } else {
        return super.getTerminal(terminalName);
62        }
63    }
64
65    void draw() {
66        fill(color(255, 0, 0));
67        rect(x, y - (level.read() * 12) / 10, 3, (level.read() * 12) / 10);
68    }
69 }
```