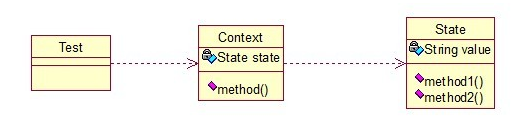
# 状态（State）设计模式

状态（State）模式核心思想就是：当对象的状态改变时，同时改变其行为，很好理解！就拿QQ来说，有几种状态，在线、隐身、忙碌等，每个状态对应不同的操作，而且你的好友也能看到你的状态，所以，状态模式就两点：1、可以通过改变状态来获得不同的行为。2、你的好友能同时看到你的变化。看图：



其中，State类是个状态类，Context类可以实现切换，我们来看看代码：

*/\*\**

*\* @Title: 状态类的核心类*

*\* @author: 夜阑珊*

*\*/*

public class State {

private String value;

public String getValue() {

return value;

}

public void setValue(String *value*) {

this.value = value;

}

public void method1() {

System.out.println("execute the first opt!");

}

public void method2() {

System.out.println("execute the second opt!");

}

}

*/\*\**

*\* @Title: 状态模式的切换类*

*\* @author: 夜阑珊*

*\*/*

public class Context {

private State state;

public Context(State *state*) {

this.state = state;

}

public State getState() {

return state;

}

public void setState(State *state*) {

this.state = state;

}

public void method() {

if (state.getValue().equals("state1")) {

state.method1();

} else if (state.getValue().equals("state2")) {

state.method2();

}

}

}

测试类：

public class Test {

public static void main(String[] *args*) {

State state = new State();

Context context = new Context(state);

*//设置第一种状态*

state.setValue("state1");

context.method();

*//设置第二种状态*

state.setValue("state2");

context.method();

}

}

输出：

execute the first opt!  
execute the second opt!

根据这个特性，状态模式在日常开发中用的挺多的，尤其是做网站的时候，我们有时希望根据对象的某一属性，区别开他们的一些功能，比如说简单的权限控制等。