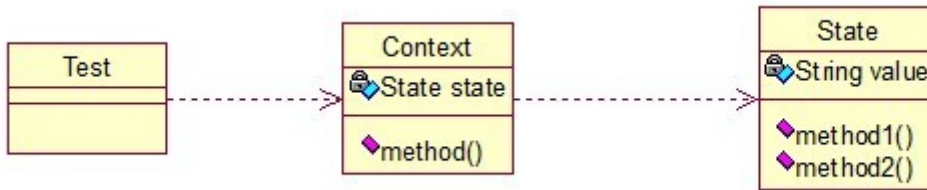


## 状态模式 (State)

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：



State类是个状态类，Context类可以实现切换，我们来看看代码：

[java] [view plaincopy](#)

```
1. package com.xtfggef.dp.state;
2.
3. /**
4.  * 状态类的核心类
5.  * 2012-12-1
6.  * @author erqing
7.  *
8.  */
9. public class State {
10.
11.     private String value;
12.
13.     public String getValue() {
14.         return value;
15.     }
16.
17.     public void setValue(String value) {
18.         this.value = value;
19.     }
20.
21.     public void method1() {
22.         System.out.println("execute the first opt!");
23.     }
24.
25.     public void method2() {
26.         System.out.println("execute the second opt!");
27.     }
28. }
```

[java] [view plaincopy](#)

```
1. package com.xtfggef.dp.state;
2.
3. /**
4.  * 状态模式的切换类    2012-12-1
5.  * @author erqing
6.  *
7.  */
8. public class Context {
9.
10.     private State state;
11.
12.     public Context(State state) {
13.         this.state = state;
14.     }
15.
16.     public State getState() {
17.         return state;
18.     }
19.
20.     public void setState(State state) {
21.         this.state = state;
22.     }
23.
24.     public void method() {
25.         if (state.getValue().equals("state1")) {
26.             state.method1();
27.         } else if (state.getValue().equals("state2")) {
28.             state.method2();
29.         }
30.     }
31. }
```

测试类:

[java] [view plaincopy](#)

```
1. public class Test {
2.
3.     public static void main(String[] args) {
4.
5.         State state = new State();
6.         Context context = new Context(state);
7.
8.         //设置第一种状态
9.         state.setValue("state1");
10.        context.method();

```

```
11.  
12.      //设置第二种状态  
13.      state.setValue("state2");  
14.      context.method();  
15.  }  
16. }
```

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。