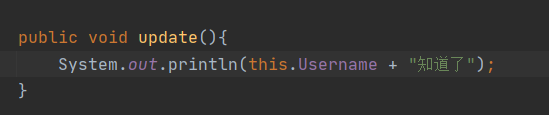
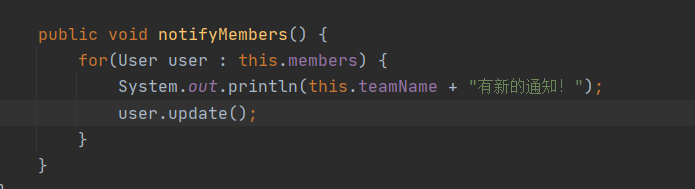
本次所采用的设计模式主要有2种。

第一种：观察者模式

在User类跟Team类之间存在观察与被观察的关系，观察者为team中的成员，被观察对象为team，现在有以下场景，队伍有了设置了新的通知，要通知队伍中的所有成员，这时候观察者模式就显现出来了优点。主要源码如下，User类实现了一个观察者的接口，Team类中定义一个方法notifymembers（），调用所有成员的更新方法。

代码截图如下：





设计模式主要优点：

1、观察者和被观察者是抽象耦合的。

2、建立一套触发机制。

第二种设计模式：外观模式

隐藏系统的复杂性，并向客户端提供了一个客户端可以访问系统的接口。

这里我们采用teamservice类来封装对于team对象的各种操作，dao类封装数据库操作，提供给客户端的只有service接口。





主要优点： 1、减少系统相互依赖。

2、提高灵活性。

3、提高了安全性。