## 第十三章 外观模式

1. **已知某子系统为外界提供功能服务，但该子系统存在很多粒度十分小的类，不便被外界系统该直接使用，采用（A）设计模式可以定义一个高层接口，这个接口使得这一子系统更加容易使用。**

A.Façade(外观) B.Singleton(单例) C.Participant(参与者) D.Decorator(装饰)

1. **下图是（D）模式实例的结构图。**



A.桥接(Bridge) B.工厂方法(Factory Method)

C.模板方法(Template) D.外观(Facade)

1. **以下关于外观模式的叙述错误的是（C）**
   1. 在外观模式中，一个子系统的外部与其内部的通信可以通过一个统一的外观对象进行
   2. 在增加外观对象之后，客户类只需要直接和外观对象交互即可，与子系统类之间的复杂引用关系有外观类对象来实现，降低了系统的耦合度
   3. 外观模式可以很好的限制客户类使用子系统，对客户类访问子系统做限制可以提高系统的灵活性
   4. 可以为一个系统提供多个外观类