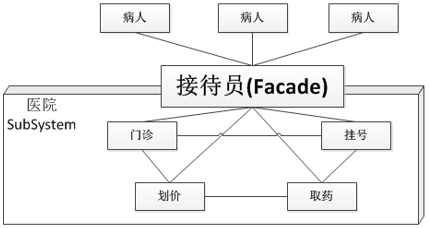
* 设计模式 – 门面模式（Facade）：

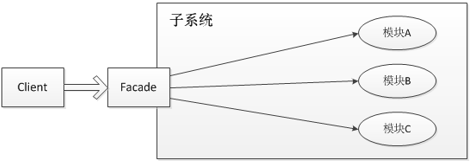
外部与一个子系统的通信必须通过一个统一的门面对象进行。门面模式提供一个高层次的接口，使得子系统更易于使用。每一个子系统只有一个门面类，但整个系统可以有多个门面类。



门面模式角色：

**1）门面（Facade）角色：**客户端可以调用这个角色的方法。此角色知晓相关子系统的功能和责任。会将所有从客户端发来的请求委派到相应的子系统去。

2）**子系统（Subsystem）角色**：可以同时有一个或多个子系统。每个子系统都不是单独的类，而是一个类的集合。每个子系统都可以被客户端调用或者被门面角色调用。子系统不知道门面角色的存在，对于子系统而言，门面仅仅是另外一个客户端而已。



* 设计模式 – 门面模式**适用性：**

　 1．为一个复杂子系统提供一个简单接口。

2．提高子系统的独立性。

3．在层次化结构中，可以使用Facade模式定义系统中每一层的入口。

**Facade设计模式更注重从架构的层次去看整个系统，而不是单个类的层次。Facade很多时候更是一种架构设计模式。**

* 设计模式 – 门面模式Demo：

