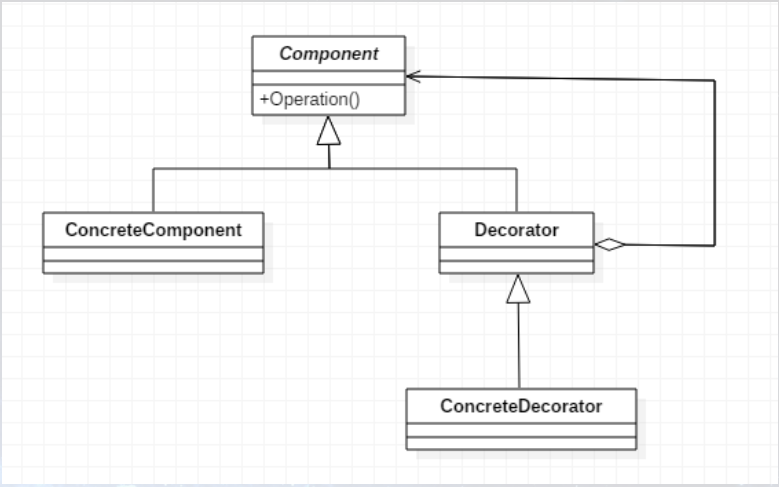
1.工厂模式：提供了一种创建对象的最佳方式。在工厂模式中，我们在创建对象时不会对客户端暴露创建逻辑，并且是通过使用一个共同的接口来指向新创建的对象。个人通俗地理解，获取指定具体工厂类的对象时可以输入指定参数向一大工厂类获取。

2.静态代理模式：分为真实角色，代理角色；真实角色和代理角色要实现同一个接口，代理角色要持有真实角色的引用。好处（个人通俗理解）：当我们需要拓展真实角色的其他功能时，无需改动真实角色的源代码，只需在代理角色中添加即可。

3.装饰模式：

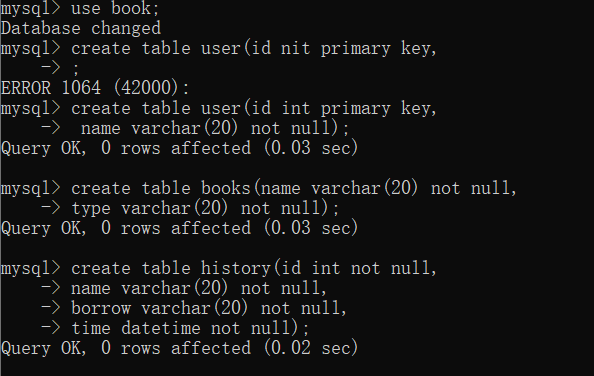


Component：是一个接口或是抽象类，就是定义我们最核心的对象，也就是最原始的对象。

ConretetComponent类：具体构件，通过继承实现Component抽象类中的抽象方法。是最核心、最原始、最基本的接口或抽象类的实现，我们要装饰的就是它。

Decorator装饰类：一般是一个抽象类，在其属性里必然有一个private变量指向Component抽象构件。

ConcreteDecorator类：可以写多个具体实现类，把最核心的、最原始的、最基本的东西装饰成其它东西。



遇到问题：对设计模式不熟悉，几种设计模式比较混乱。不大会设计关系数据库。

解决方法：通过B站的教学视频，及网上的代码讲解，教程对设计模式有了较好的理解。