1. 策略模式

有很多不同的策略，都实现了一个共同的虚基类，每个策略都有独特的操作。然后有一个管理这些策略的类（锦囊），在每次使用策略前都需要将策略放入到锦囊中。策略模式实际上就是使用不同策略的过程。

1. 代理模式

明星与经纪人的关系。经纪人和明星都继承自一个共同的基类（人），经纪人可以代理明星与一些合作商进行谈判，经纪人做的事情都是替明星做的。也就是看似是经纪人在做，而实际上是明星在做。

1. 单例模式

单例模式很简单，就是一个类只有一个实例，任何时候获得这个类的实例时都是获得这个唯一的实例。这种模式适用于程序中仅仅只需要类的一个实例的情况。

1. 多例模式

多例模式实际上是单例模式的一个变种，在程序中需要保存类的多个实例的情况下使用。

1. 工厂模式

顾名思义，工厂模式就是建立一个工厂，然后给定一些原材料（或说是条件）生成出特定的产品。

1. 抽象工厂模式

工厂模式只是单层继承，而抽象工厂模式则是多层继承。多层继承的好处在于在更高层的类集合中可以包括更多的属性。比如最底层的虚基类是人，第二层抽象类是黄种人、白种人和黑种人，第三层可以是女性黄种人、女性白种人、女性黑种人、男性黄种人、男性白种人、男性黑种人。

1. 门面模式

将很多用户的一些共同行为抽象出来形成一个方法，简便用户的操作的模式。比如人们去邮局寄信，总是会进行如下四步：写信、写收件人地址和信息、将信放进信封中，寄信。而我们可以将这四个步骤抽象出来，形成一个寄信的方法，只要用户将信的内容和收件人的地址和信息交给邮局，邮局就可以代替用户完成所有寄信的操作。