1. 模式定义

原型模式(Prototype Pattern)：原型模式是一种对象创建型模式，用原型实例指定创建对象的种类，并且通过复制这些原型创建新的对象。

原型模式允许通过一个原型对象创建一个或多个同类型的其他对象，而无须知道任何创建的细节



**二、**模式动机

复制一个对象，从而克隆出多个与原型对象一模一样的对象——原型模式

有些对象的创建过程较为复杂，而且需要频繁创建

通过给出一个原型对象来指明所要创建的对象的类型，然后用复制这个原型对象的办法创建出更多同类型的对象

三、例子

例复制自己，比较少用，很多类提供clone方法。例如CTRL+C /CTRL+V,两个对象内容相同，但内存地址不相同。

四、优缺点

优点：

* 简化对象的创建过程，通过复制一个已有实例可以提高新实例的创建效率
* 扩展性较好
* 简化创建结构，原型模式中产品的复制是通过封装在原型类中的克隆方法实现的，无须专门的工厂类来创建产品
* 可以使用深克隆的方式保存对象的状态，以便在需要的时候使用，可辅助实现撤销操作

缺点：

* 需要为每一个类配备一个克隆方法，而且该克隆方法位于一个类的内部，当对已有的类进行改造时，需要修改源代码，违背了开闭原则
* 在实现深克隆时需要编写较为复杂的代码，而且当对象之间存在多重的嵌套引用时，为了实现深克隆，每一层对象对应的类都必须支持深克隆，实现起来可能会比较麻烦

五、示例代码