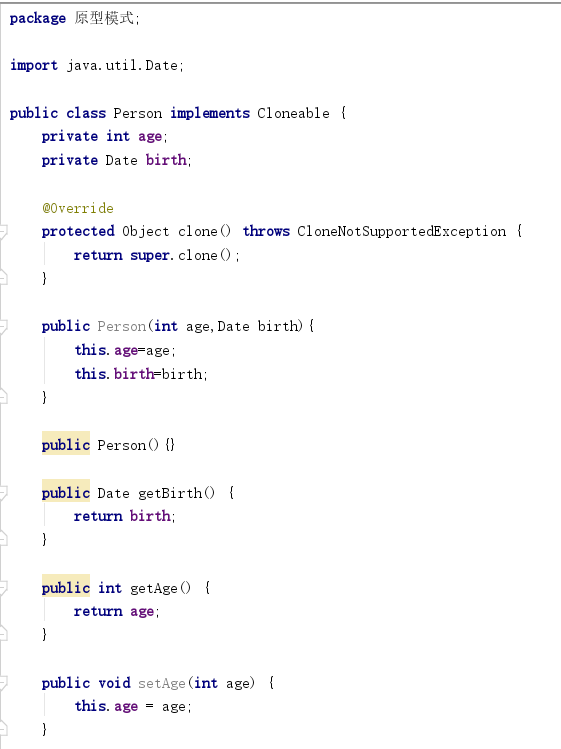
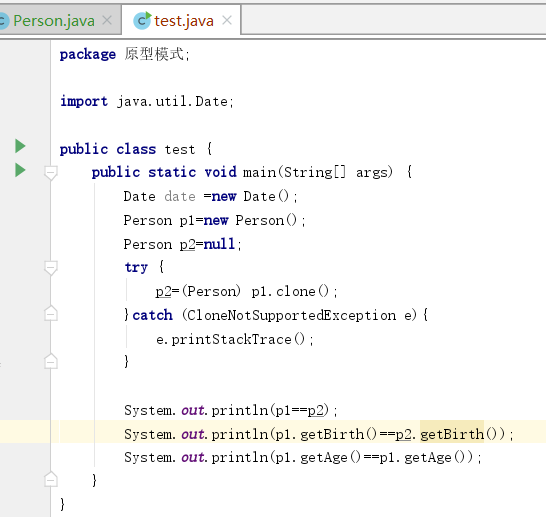
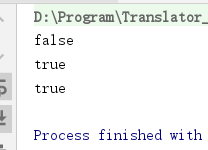
# 定义

原型模式是一种创建型模式，拜托了类的构造模式，原型模式告诉我们，想要创建一个对象，我们不必关心对象的具体类型，而是找到一个对象，然后通过克隆来创建一个一模一样的对象。

# **浅拷贝和深拷贝**

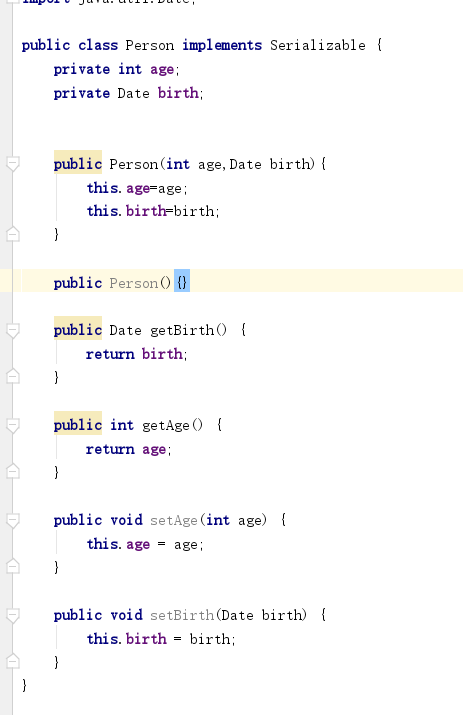


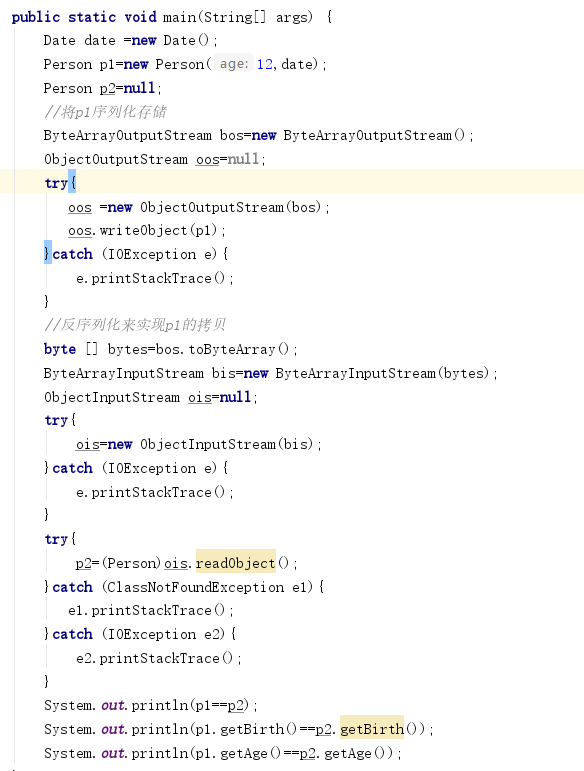


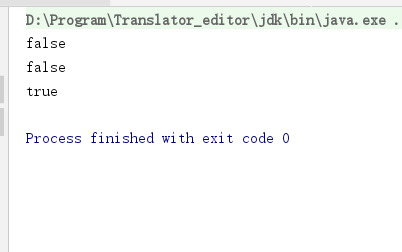


从结果中我们可以知道，这种克隆形式真正克隆了Person对象，但是其内部包含的其他对象(非基本变量类型)却还是引用被克隆对象的，所以不是真正意义上的克隆。

1. **利用序列化深度克隆**







可以看到p2的内部对象是完全被克隆而不是引用p1的。

1. 应用场景

原型模式一般很少单独出现，一般都是和工厂方法模式一起搭配使用，通过clone来创建对象，然后由工厂方法返回。依赖倒置原则提醒我们创建对象时尽量不要依赖具体的对象类型，原型模式就很好印证了这句话，避免僵硬的使用new来进行对象创建。

1. 优缺点

优点:

\* 向客户隐藏实例生成的细节

\* 某些环境下啊，复制对象比新创建对象更有效

\* 提供让客户自主创建未知类型对象的方法

\* 减少子类的构造，原型模式通过克隆而不是工厂方法来产生一个对象

缺点:

\* 对象复制有时比较复杂，特别是对象层级嵌套很深时。