

## 设计模式：简单工厂、工厂方法、抽象工厂之小结与区别

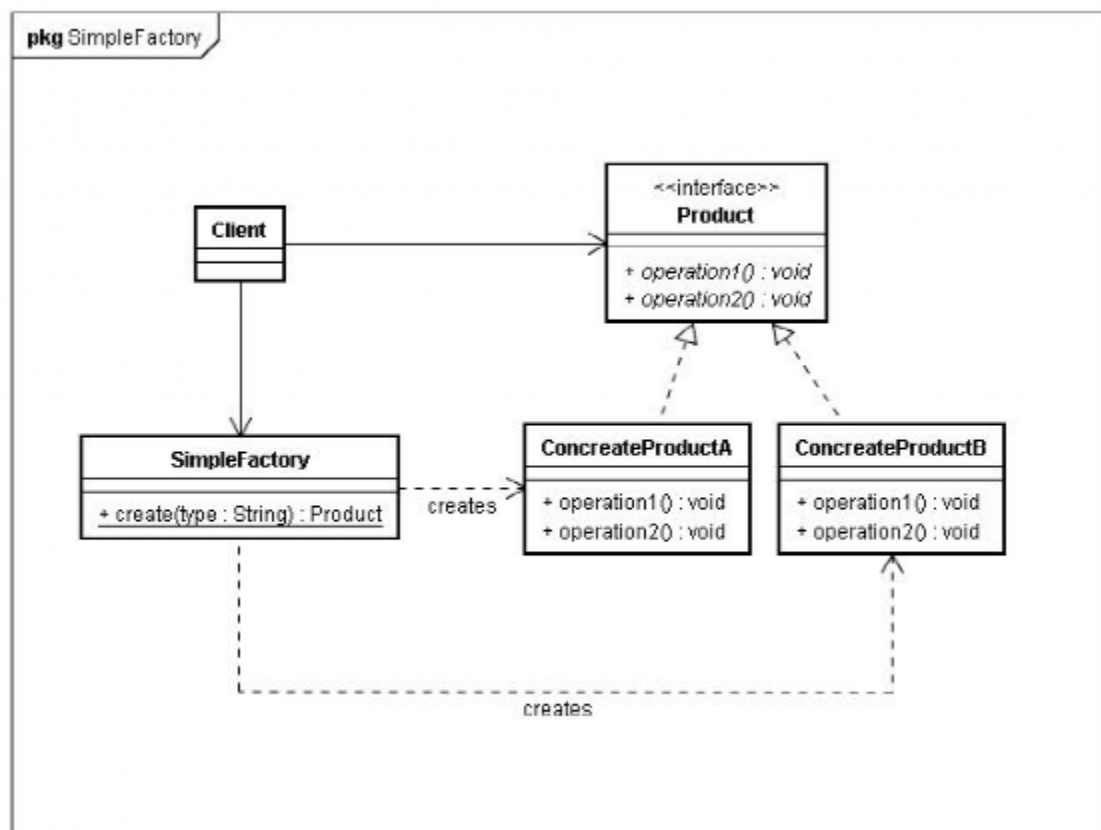
简单工厂，工厂方法，抽象工厂都属于设计模式中的创建型模式。其主要功能都是帮助我们  
把对象的实例化部分抽取了出来，优化了系统的架构，并且增强了系统的扩展性。

本文是本人对这三种模式学习后的一个小结以及对他们之间的区别的理解。

### 简单工厂

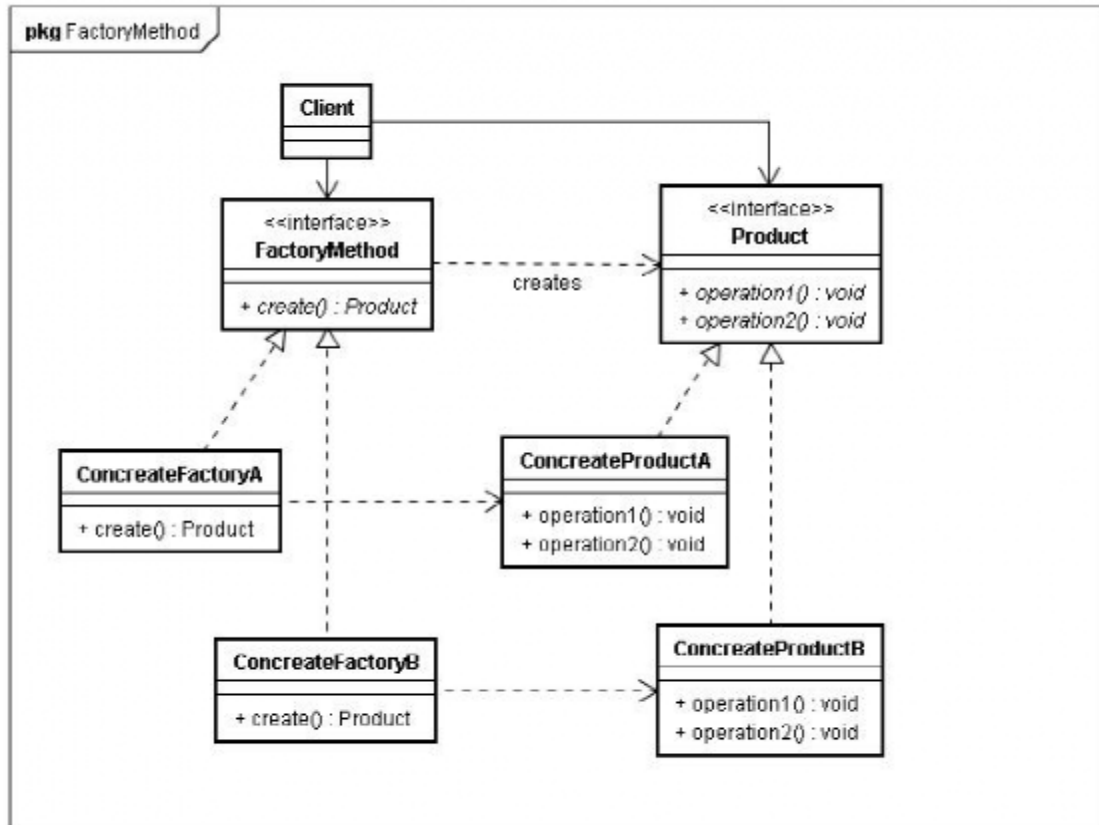
简单工厂模式的工厂类一般是使用静态方法，通过接收的参数不同来返回不同的对象实例。

不修改代码的话，是无法扩展的。



## 工厂方法

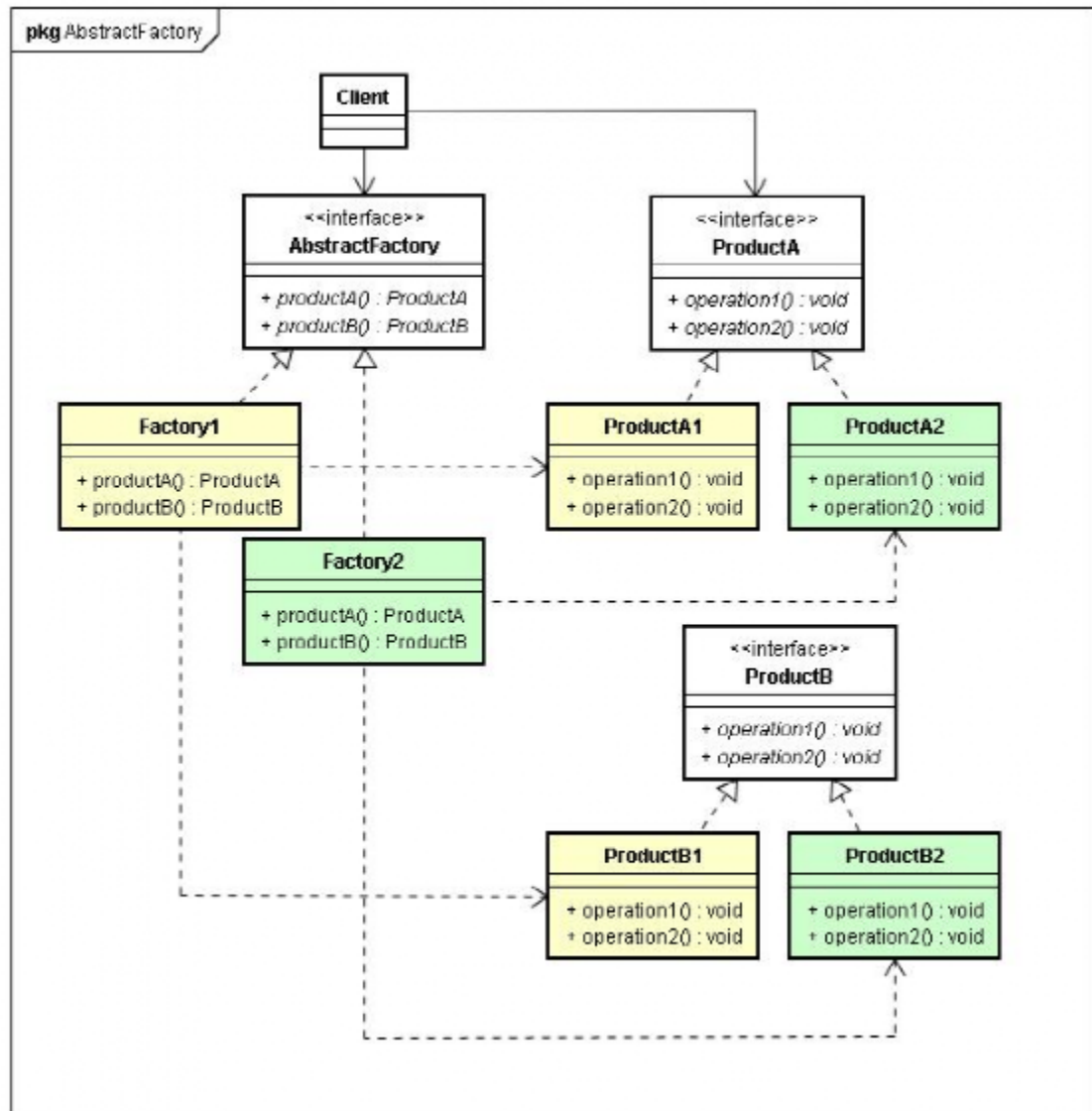
工厂方法是针对每一种产品提供一个工厂类。通过不同的工厂实例来创建不同的产品实例。在同一等级结构中，支持增加任意产品。



## 抽象工厂

抽象工厂是应对产品族概念的。比如说，每个汽车公司可能要同时生产轿车，货车，客车，那么每一个工厂都要有创建轿车，货车和客车的方法。

应对产品族概念而生，增加新的产品线很容易，但是无法增加新的产品。



## 小结

★工厂模式中，重要的是工厂类，而不是产品类。产品类可以是多种形式，多层继承或者是单个类都是可以的。但要明确的，工厂模式的接口只会返回一种类型的实例，这是在设计产品类的时候需要注意的，最好是有父类或者共同实现的接口。

★使用工厂模式，返回的实例一定是工厂创建的，而不是从其他对象中获取的。

★工厂模式返回的实例可以不是新创建的，返回由工厂创建好的实例也是可以的。

## 区别

简单工厂： 用来生产同一等级结构中的任意产品。（对于增加新的产品，无能为力）

工厂方法： 用来生产同一等级结构中的固定产品。（支持增加任意产品）

抽象工厂： 用来生产不同产品族的全部产品。（对于增加新的产品，无能为力；支持增加产品族）

以上三种工厂 方法在等级结构和产品族这两个方向上的支持程度不同。所以要根据情况考虑应该使用哪种方法。