# 简单工厂、工厂方法、抽象工厂

之前写过简单工厂，有时候面试的时候会被问起这三个，本文就写写。

先说一下，简单工厂里，创建实例的那个方法是static的就是静态工厂，当然也可以不用static。

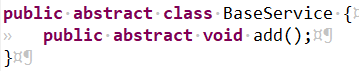
简单工厂好处是调用者与实现类的硬编码，解耦了。缺点是违反开闭原则——可以扩展，但是不能修改，简单工厂正好反过来，要想再多加一个RoleService，必须修改ServiceFactory方法，在里面加一个if分支。

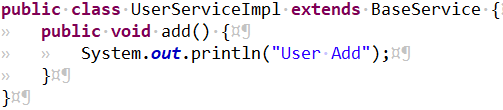
而且你想想，要是Service多了，这个Factory类里面得有多少分支。

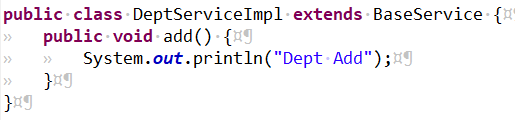
## 工厂方法

如果说简单工厂是一个工厂创建N个对象，那么工厂方法就是一个工厂创建一个对象。

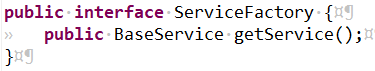
需要实例的对象及父类

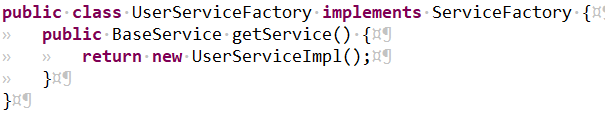


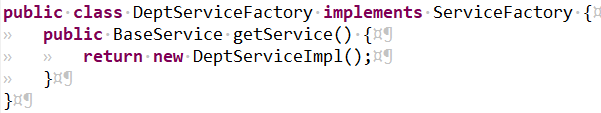




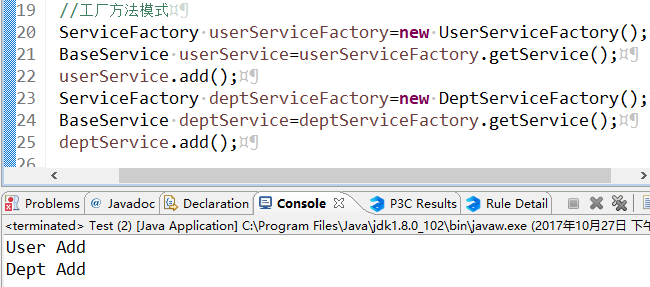
工厂相关类







测试：

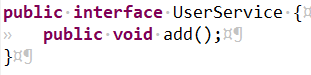


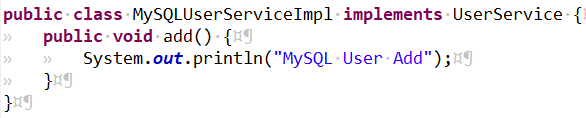
这种写法很符合开闭原则吧，如果我们要新增加一个RoleService，不用修改现有任何代码，照方抓药，新建RoleServiceImpl，RoleServiceFactory就可以了。

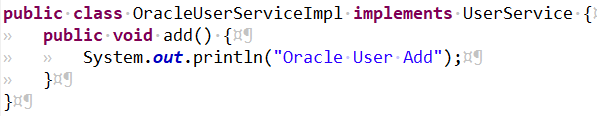
## 抽象工厂

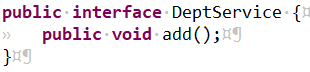
假设我们有两个数据源：Oracle、MySQL，每个数据库都有User、Dept的操作，按照抽象工厂，我们可以这么做。

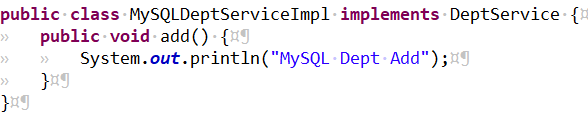
需要实例的对象及接口：

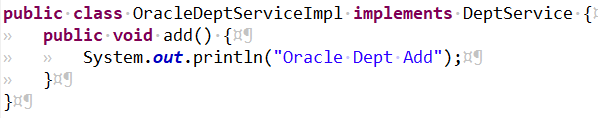




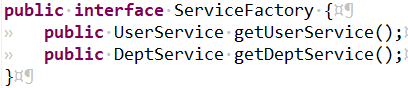


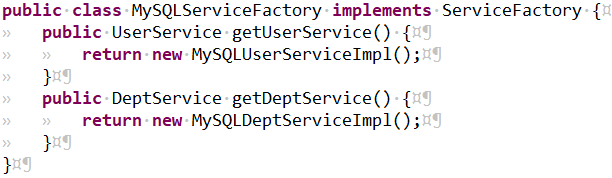


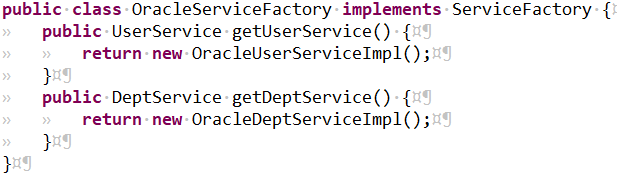




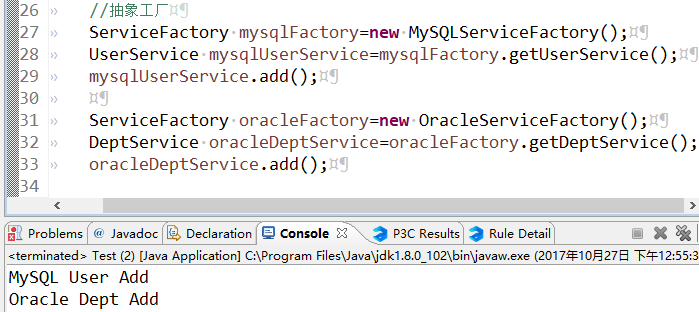
工厂相关：







测试：



这个好处不言而喻了吧，比如我们要增加一个SQLServer的数据源，不用修改原来的代码，照方抓药，增加实现和Factory就行了。

缺点就是想在原来的地方增加RoleService这样的，就得改代码了。