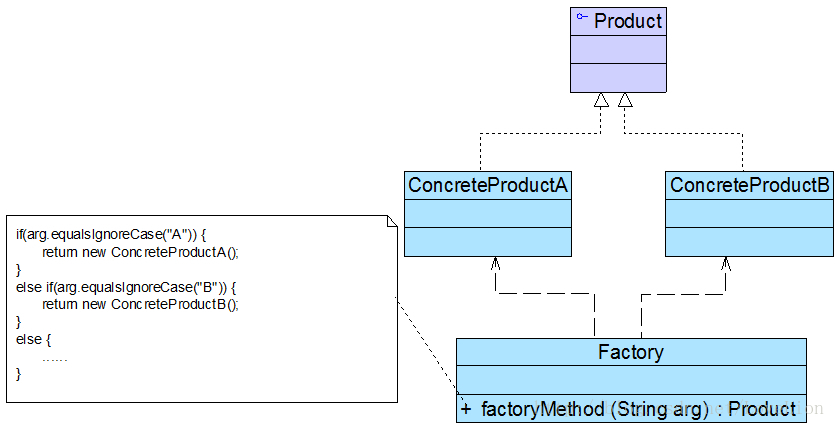
简单工厂模式(Simple Factory Pattern)：定义一个工厂类，它可以根据参数的不同返回不同类的实例，被创建的实例通常都具有共同的父类。因为在简单工厂模式中用于创建实例的方法是静态(static)方法，因此简单工厂模式又被称为静态工厂方法(Static Factory Method)模式，**它属于类创建型模式**。

简单工厂模式的要点在于：**当你需要什么，只需要传入一个正确的参数，就可以获取你所需要的对象，而无须知道其创建细节**。简单工厂模式结构比较简单，其核心是工厂类的设计，其结构如图所示：

在简单工厂模式结构图中包含如下几个角色：

       ● **Factory（工厂角色）**：工厂角色即工厂类，它是简单工厂模式的核心，负责实现创建所有产品实例的内部逻辑；工厂类可以被外界直接调用，创建所需的产品对象；在工厂类中提供了静态的工厂方法factoryMethod()，它的返回类型为抽象产品类型Product。

       ● **Product（抽象产品角色）**：它是工厂类所创建的所有对象的父类，封装了各种产品对象的公有方法，它的引入将提高系统的灵活性，使得在工厂类中只需定义一个通用的工厂方法，因为所有创建的具体产品对象都是其子类对象。

       ● **ConcreteProduct（具体产品角色）**：它是简单工厂模式的创建目标，所有被创建的对象都充当这个角色的某个具体类的实例。每一个具体产品角色都继承了抽象产品角色，需要实现在抽象产品中声明的抽象方法。

改进：

Sunny软件公司开发人员发现在创建具体Chart对象时，每更换一个Chart对象都需要修改客户端代码中静态工厂方法的参数，客户端代码将要重新编译，这对于客户端而言，违反了“开闭原则”，有没有一种方法能够在不修改客户端代码的前提下更换具体产品对象呢？答案是肯定的，下面将介绍一种常用的实现方式。

       我们可以将静态工厂方法的参数存储在XML或properties格式的配置文件中，如下config.xml所示：

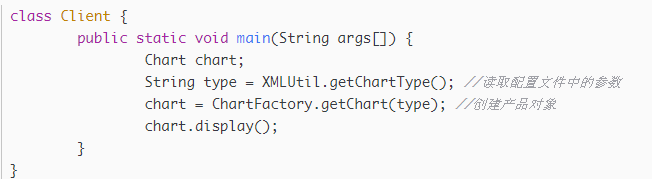
<?xml version="1.0"?>

<config>

<chartType>histogram</chartType>

</config>

再通过一个工具类XMLUtil来读取配置文件中的字符串参数，XMLUtil类的代码如下所示：

在引入了配置文件和工具类XMLUtil之后，客户端代码修改如下：

不难发现，在上述客户端代码中不包含任何与具体图表对象相关的信息，如果需要更换具体图表对象，只需修改配置文件config.xml，无须修改任何源代码，符合“开闭原则”。

简单工厂模式总结：

优点：

(1) 工厂类包含必要的判断逻辑，可以决定在什么时候创建哪一个产品类的实例，客户端可以免除直接创建产品对象的职责，而仅仅“消费”产品，简单工厂模式**实现了对象创建和使用的分离**。

       (2) 客户端无须知道所创建的具体产品类的类名，只需要知道具体产品类所对应的参数即可，对于一些复杂的类名，通过简单工厂模式可以在一定程度减少使用者的记忆量。

       (3) 通过引入配置文件，可以在不修改任何客户端代码的情况下更换和增加新的具体产品类，在一定程度上提高了系统的灵活性。

缺点：

(1) 由于工厂类**集中了所有产品的创建逻辑，职责过重，一旦不能正常工作，整个系统都要受到影响**。

       (2) 使用简单工厂模式势必会增加系统中类的个数（引入了新的工厂类），增加了系统的复杂度和理解难度。

       (3) 系统扩展困难，一旦添加新产品就不得不修改工厂逻辑，在产品类型较多时，有可能造成工厂逻辑过于复杂，不利于系统的扩展和维护。

       (4) 简单工厂模式由于使用了静态工厂方法，造成工厂角色无法形成基于继承的等级结构。

       (5) 最大的缺点是当有新产品要加入到系统中时，必须修改工厂类，那么就意味着违反“开闭原则”。

适用场景：

(1) 工厂类负责创建的对象比较少，由于创建的对象较少，不会造成工厂方法中的业务逻辑太过复杂。

(2) 客户端只知道传入工厂类的参数，对于如何创建对象并不关心。

[简单工厂模式](https://blog.csdn.net/LoveLion/article/details/17517213)

[工厂三兄弟之简单工厂模式（一）：图表库的设计](https://blog.csdn.net/lovelion/article/details/9300337)

[工厂三兄弟之简单工厂模式（二）：简单工厂模式概述](https://blog.csdn.net/lovelion/article/details/9300549)

[工厂三兄弟之简单工厂模式（三）：图表库的简单工厂模式解决方案](https://blog.csdn.net/lovelion/article/details/9300657)

[工厂三兄弟之简单工厂模式（四）：图表库解决方案的改进，简单工厂模式的简化，简单工厂模式总结](https://blog.csdn.net/lovelion/article/details/9300731)