配置文件代码

```xml
<?xml version="1.0"?>
<config>
  <className>CashPayFactory</className>
</config>
```

工具类XMLUtil代码片段

```java
//创建DOM文档对象
DocumentBuilderFactory dFactory = DocumentBuilderFactory.newInstance();
DocumentBuilder builder = dFactory.newDocumentBuilder();
Document doc;
doc = builder.parse(new File("config.xml"));
//获取包含类名的文本节点
NodeList nl = doc.getElementsByTagName("className");
Node classNode = nl.item(0).getFirstChild();
String cName = classNode.getNodeValue();
//通过类名生成实例对象并将其返回
Class c = Class.forName(cName);
Object obj = c.newInstance();
return obj;
```

抽象工厂类代码

```java
public abstract class PayMethodFactory{
  public abstract AbstractPay getPayMethod();
}
```

具体工厂类代码

```java
public class CashPayFactory extends PayMethodFactory{
  public AbstractPay getPayMethod(){
    return new CashPay();
  }
}
```

修改后的客户类代码片段

```java
PayMethodFactory factory;
AbstractPay payMethod;
//getBean()的返回类型为Object, 此处需要进行强制类型转换
factory = (PayMethodFactory)XMLUtil.getBean();
payMethod = factory.getPayMethod();
payMethod.pay();
```