Java序列化

Java序列化是为了Java的RMI调用和javaBeans的配置。

要序列化一个对象，需要该对象实现java.io.Serializable接口。这个接口里面没有任何字段和方法声明。然后创建某些OutputStream对象(因为是字节序列)，反序列化时也要创建某些InputStream对象。但如果在一个被序列化的对象中，并不想某个属性被序列化，比如密码之类的属性，可在该属性面前加上transient关键字。

加上transient关键字是一种方式，另一种方式为实现Externalizable接口。这个接口继承于java.io.Serializable接口，并添加了两个方法writeExternal和readExternal方法。这两个方法会在序列化和反序列化的时候自动调用。相当于对象实现自己的序列化，在方法中手动序列化对象想要序列化的属性。

如不想实现Externalizable接口，另一种方式就是实现Serialzable接口，并在类中添加writeObject和readObject方法。这两个方法的声明一定要为：

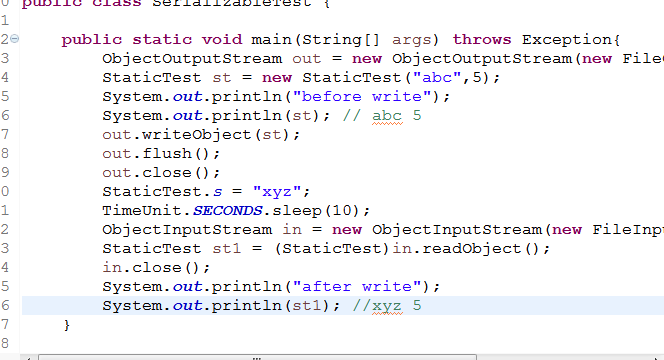
private void writeObjet(ObjectOutputStream stream) throws IOException

private void readObjet(ObjecInputStream stream) throws IOException

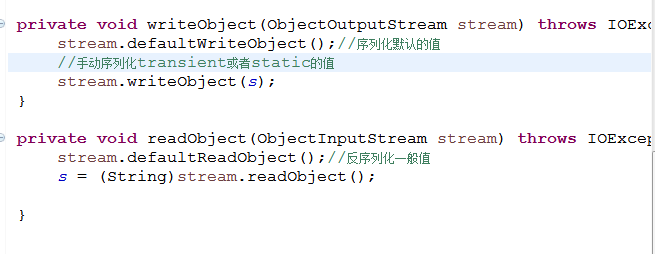
这两个方法做的事情与实现Externalable接口的writeExternal和readExternal方法一致。

ObjectOutputStream在将对象序列化时，会先检查对象是否有writeObject方法，如果有就调用该方法，没有就调用默认的方法完成操作，反之亦然。

此外static修饰符修饰的属性在序列化时也不会被序列化，必须手动在writeObject或者writeExternal方法中操作，反序列化时亦一样。如下所示：



要想序列化static值或者transient值，可采用以下方式。



Java序列化只适合于java，另一种更具互操作性的解决方案是将数据转换为XML格式。

在java中将数据转换为xml有多种方式。

1. 使用java自带的DOM和JAXB方式处理
2. 使用DOM4J处理
3. 使用XStream处理
4. 使用java api的stax来处理xml