**ResultSet不能作为返回值，请仔细阅读下面的内容**

**前提：程序结束前一定要关闭Connection和Statement，ResultSet**

**Connection关闭，ResultSet会关闭**

**Statement关闭，ResultSet会关闭**

**所以ResultSet不能作为返回值，原因有以下两点**

**①你不知道你在使用这个ResultSet时，它的Connection或者Statement会不会突然关闭，一旦关闭，ResultSet也会跟着关闭**

**②同样的，如果ResultSet作为方法返回类型，你通过这个方法得到ResultSet并使用后，就无法关闭ResultSet的Connection**

**所以查询方法尽量以Result作为返回值**

## 但是Result在javax.servlet.jar中，这里无法配置，所以我们在这里 以ArrayList的形式返回

**一，DAO模型(Data Access Object)**

**将数据库相关操作封装在一个类中**

DAO模型首先要提供一个**DAO接口**，然后提供一个**DAO实现类**，在编写一个**DAO工厂**

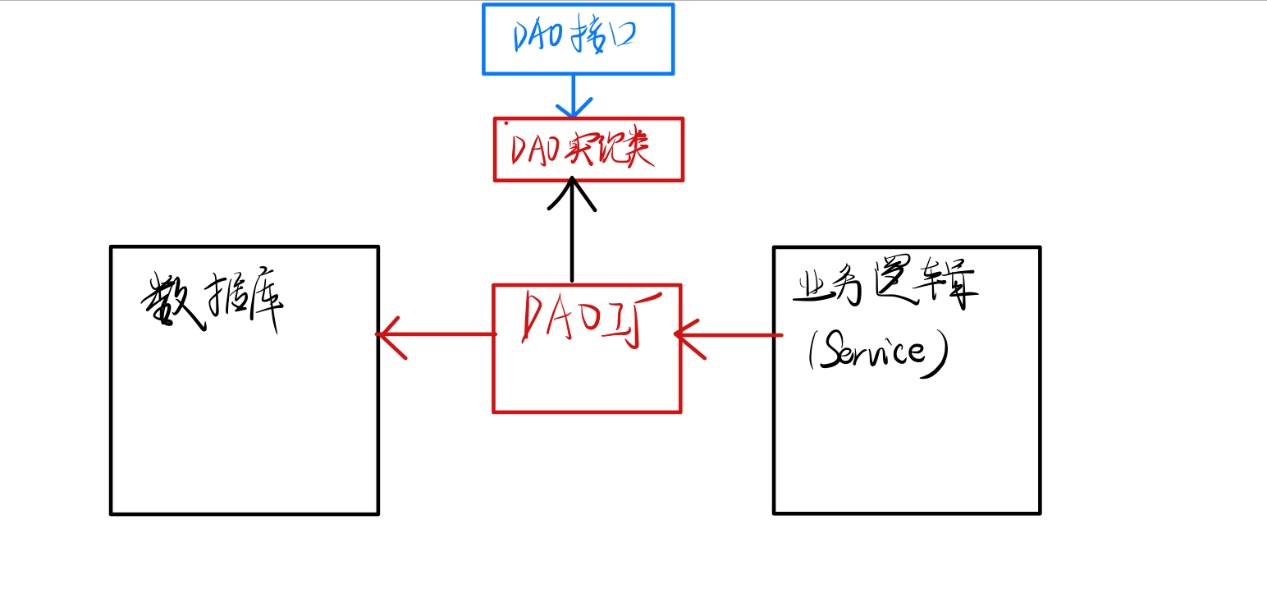
**DAO接口**：规定了DAO实现类可以进行哪些数据库访问操作

**DAO实现类**：连接数据库，对数据库相关操作进行了封装

**DAO工厂**：为Service返回一个DAO实现类对象

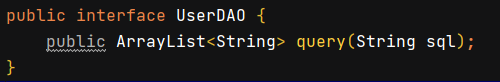
**当业务逻辑(Service)需要访问数据库时，并不是直接访问数据库，而是从DAO工厂中**

**获得一个DAO实现类对象，然后通过这个对象 对数据库进行访问**

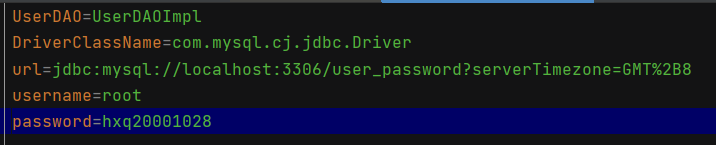


**二，DAO模型实现过程如下**

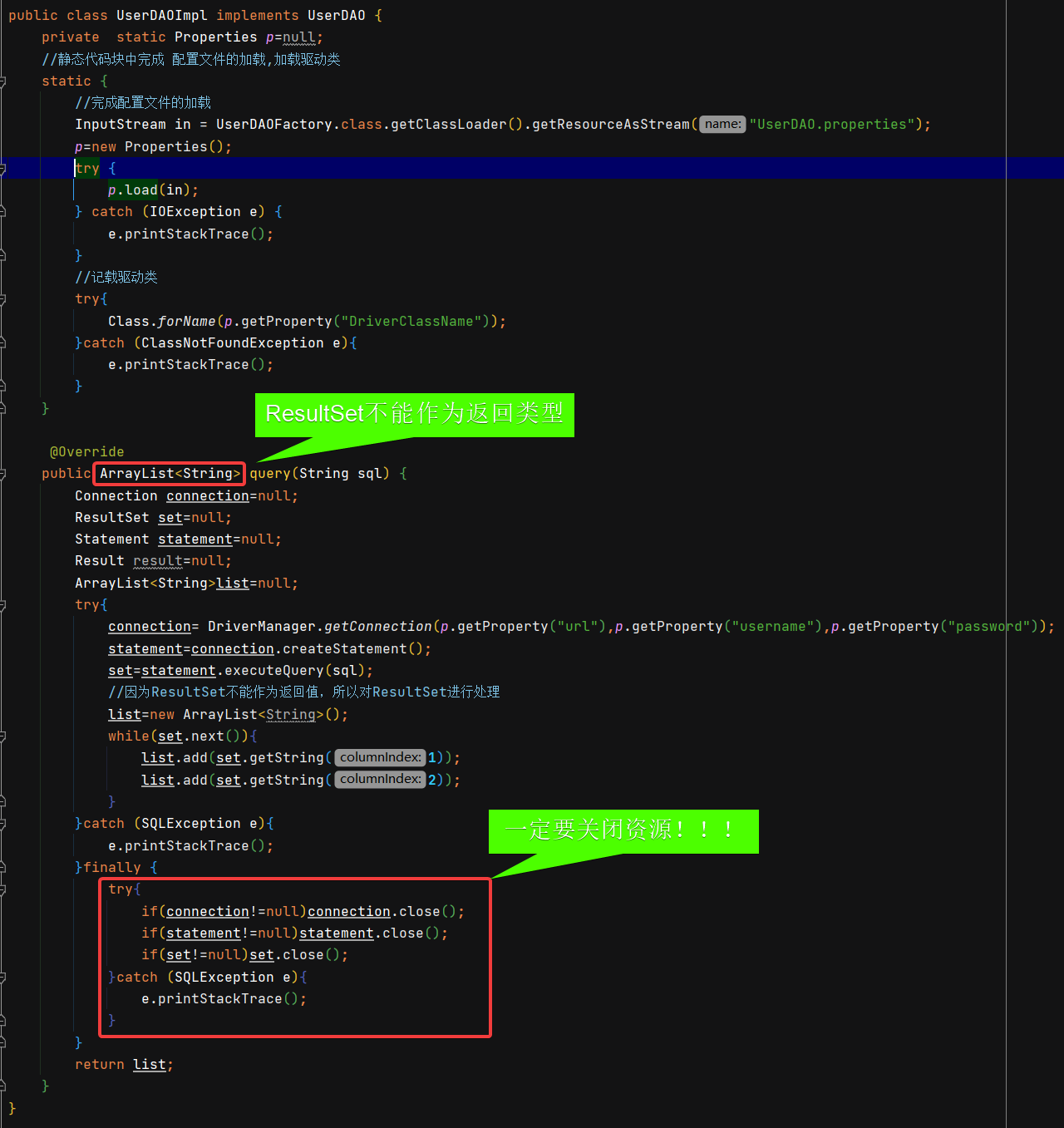
**①实现一个DAO接口——UerDAO——规定了 DAO实现类 可以进行哪些操作**



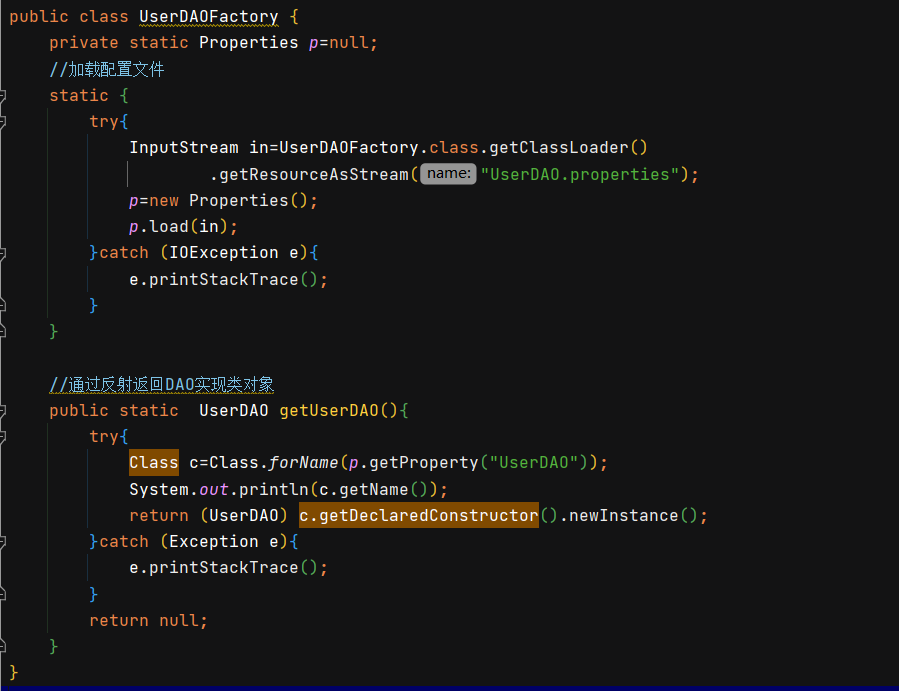
**②把Service要访问的数据库的参数，还有DAO实现类的名字 放在配置文件中**



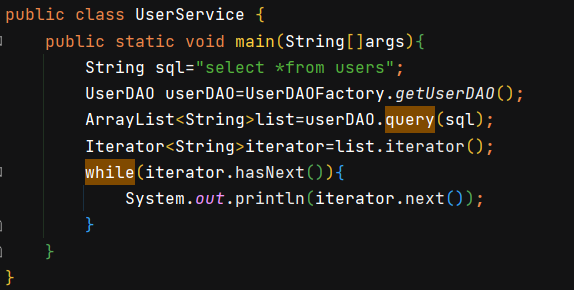
**③实现一个DAO实现类——UserDAOImpl**



**④实现DAO工厂类——UserDAOFactory**



**⑤用一个类UserService来表示Service操作**



**1，配置文件的作用，以及工厂类中为什么要使用反射来获取 DAO实现类对象**

配置文件中有——①Service要访问的数据库的配置信息 ②以及 DAO实现类的名字

①如果Service要访问别的数据库，只需要在配置文件中修改配置信息即可，不需要修改代码

②DAO工厂类通过从配置文件中获取 DAO实现类的名字，然后通过反射创建其实例对象——后期，如果我们需要其他的DAO实现类对象，我们只需要在 配置文件中修改 实现类的名字即可，不需要修改代码

**2，在DAO实现类和DAO工厂中，为什么 Properties引用要设置为静态，并且 加载配置文件 和 加载驱动类要放在静态代码块中？**

Java是多线程的编程语言，一个程序中，可能有多个线程同时要通过工厂类获得DAO实现类对象，可能有多个线程通过要使用DAO实现类来访问数据库

这样子，我们只需要一个Properties引用，配置文件也只需加载一次，驱动类也只需加载一次——节约了时间，空间

**3，一定要关闭Connection和Statement，ResultSet**——所以不能使用ResultSet作为返回类型，但我们可以用Result作为返回类型，但是由于这里的配置环境无法使用Result，我们使用ArrayList来代为处理结果

**4，注意，我们要访问数据库时——需要用到SQL包**

**但是我们应该保证 SQL包下的类（除了ResultSet）只能出现在 DAO工厂和DAO实类中，Service中不应该出现SQL包下的类**