

# 一连接

使用sqlyog时需要对数据库进行连接，指定哪个IP下的哪个数据库，输入数据库用户名和密码即可连接。

现在不是sqlyog而是使用java后台进行连接，就需要获取一个连接的对象，然后调用对象connection方法进行连接。

获取这个对象，就需要指定数据库的IP数据库，用户名，密码。

我们将IP数据库，用户名，密码，mysql驱动类路径以文件的形式（.properties）保存，然后在test方法中使用反射加载并生成一个输入流，将IP数据库，用户名，密码，驱动赋值给具体的字符串。

接下来就要具体到实例化数据库驱动对象，sun公司提供了Driver接口，让各个数据库的驱动（Driver）实现此接口给出具体的连接方法，所以实例化Driver对象需要多态。

使用反射加载mysql的驱动类，Class.forName(驱动字符串)，这时将mysql的Driver类加载到内存中。

在mysql的Driver中有静态代码块，随着类加载生效，功能为调用DriverManager类的注册方法，并实例化自己类的对象传入，此时DriverManager中静态属性就被赋值为mysql的Driver的对象。此时就相当于得到了管理mysql的驱动对象的DriverManager类，但是还没有调用connection方法连接。

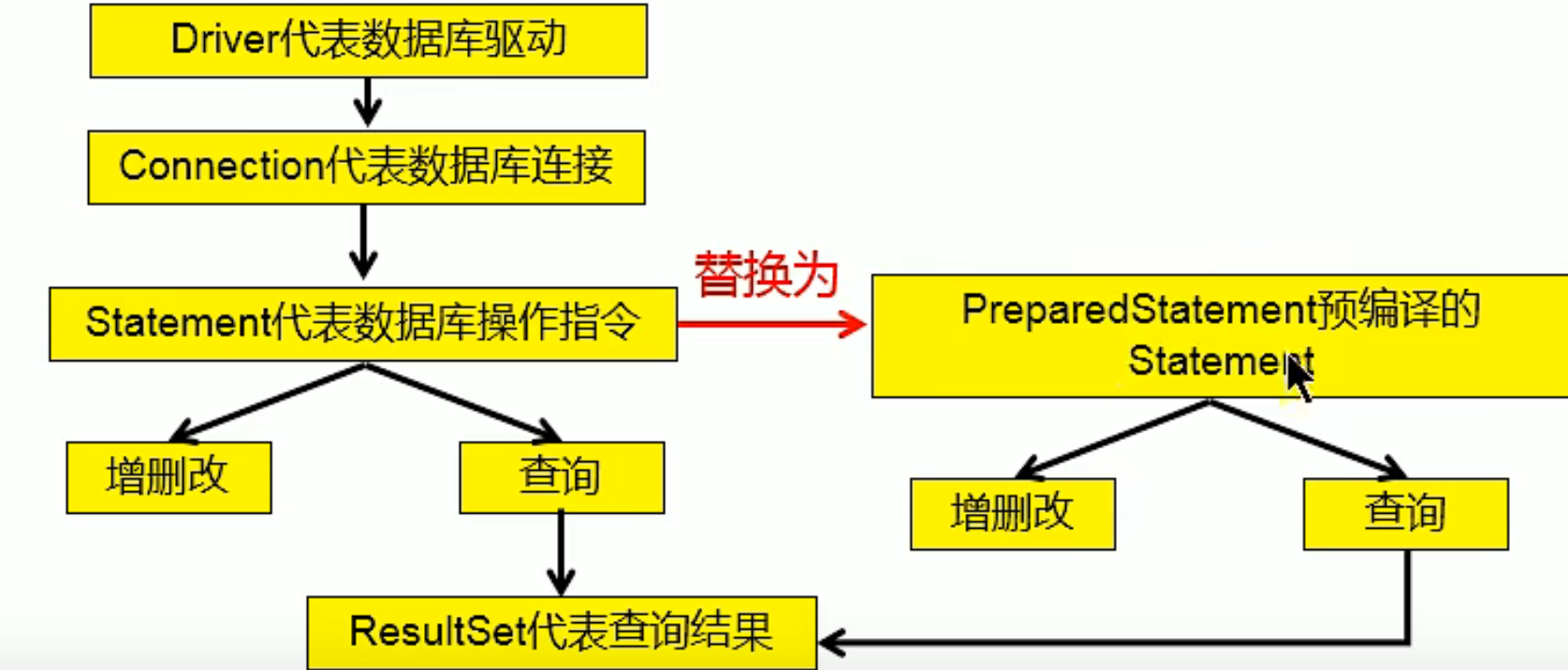
之后调用DriverManager静态方法connection()（已被注册为mysql的Driver）,将反射获取的文件中的IP数据库，用户名，密码传入connection()。此时得到的connection对象才是真正的连接对象。

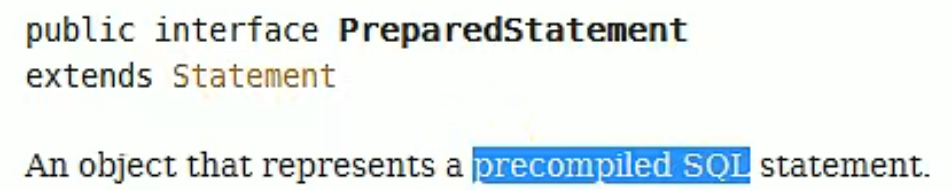
最后输出一下connection对象看是不是空指针，不是则connection连接对象实例化成功。

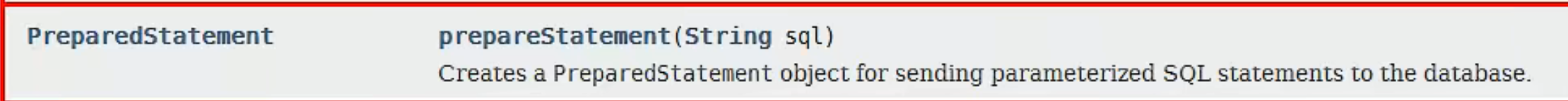




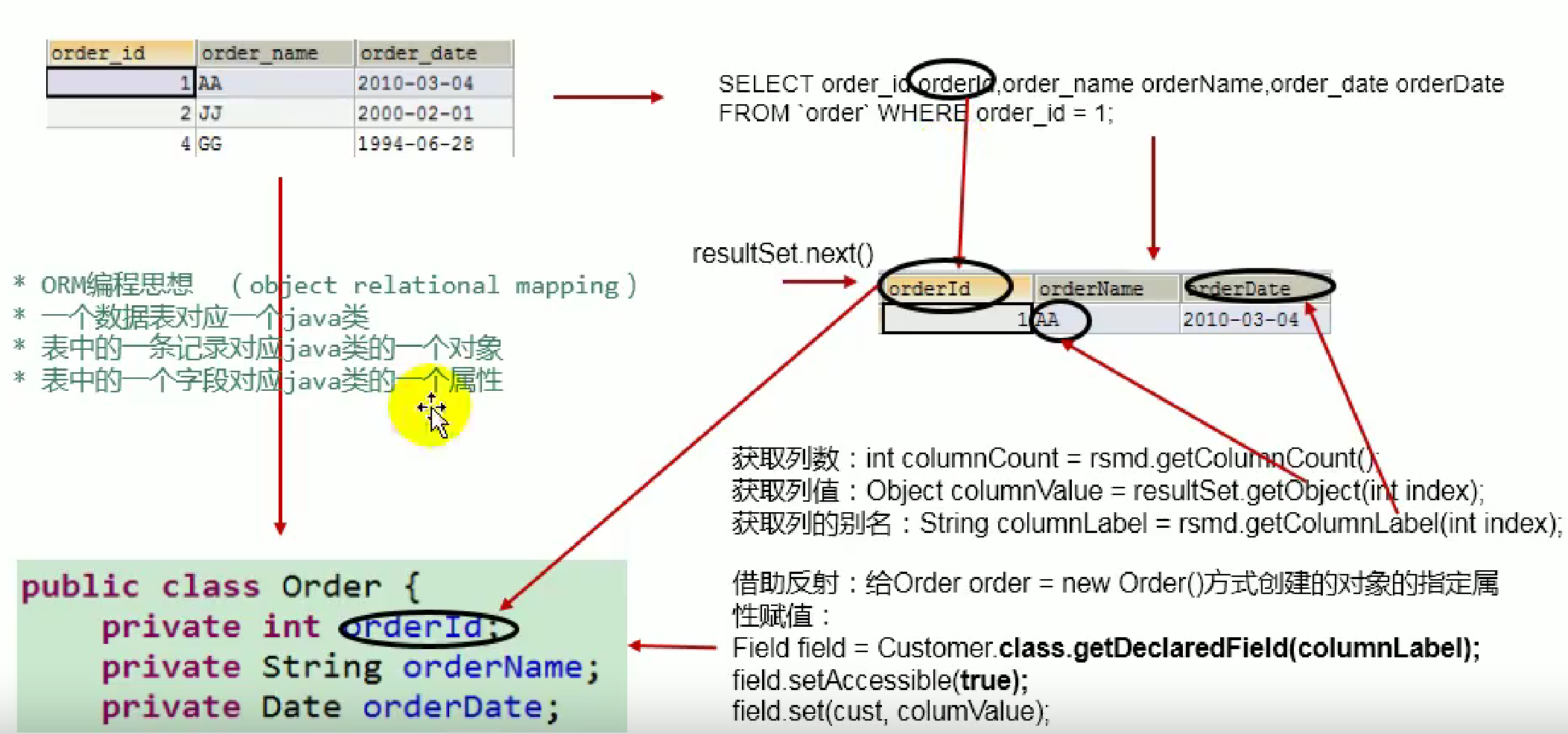
# 二PreparedStatement传输sql语句到数据库







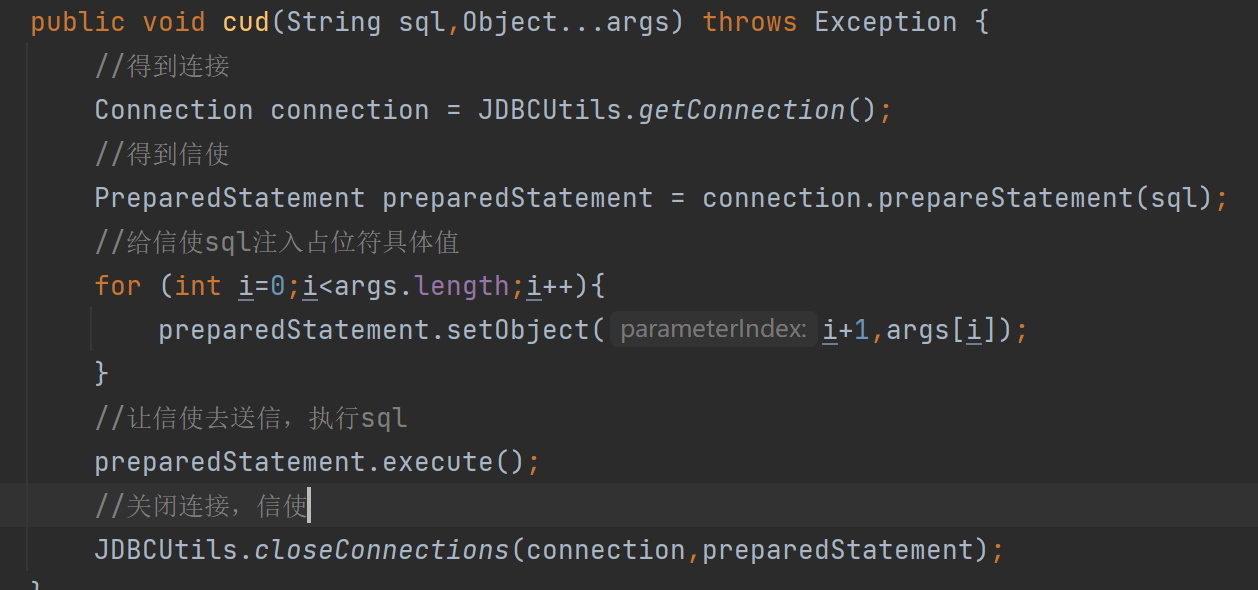
获取到连接对象之后，使用connection对象的preparestatement方法获取传输sql对象的实例。



如果说sqlyog是工具，可以让sql语句去执行，那么jdbc就是我们自己写的工具。因为后台又不是人，不会使用sqlyog。

## 单行增删改

1. 连接数据库，获取mysql连接对象
2. 预编译sql语句并使用连接对象生成preparedStatement对象，作为信使。
3. 给占位符填充信息，填充之后让信使执行excute方法去数据库传递sql语句，并执行sql。
4. 关闭连接，信使。



## 查询

1. 得到连接对象
2. 连接对象得到信使对象并传入sql
3. 给占位符注入值并执行excuteQuery，得到结果集
4. 取结果集的元数据，通过元数据得到列数count，循环结果集.next()，每次循环取出结果集中一行数据，嵌套一层循环，根据count取出每行中各个列的值和列名，根据列名使用反射得到类的属性，指定一个类的对象给属性设置值，完成内嵌循环之后，将这个对象添加到list中，完成总的循环之后，查询到的结果有几行，list中就有几个对象。
5. 关闭连接，信使，结果集

