**jdbc的认知，什么是jdbc**

JDBC（Java DataBase Connectivity,java数据库连接）是一种用于执行SQL语句的Java API，可以为多种关系数据库提供统一访问，它由一组用Java语言编写的类和接口组成。

**jdbc的开发步骤**

**(1) 、注册驱动、加载驱动**

**(2) 、获取数据库访问的连接对象**

**(3) 、获取到语句执行平台(Statement PreparedStatement)**

**(4) 、执行sql语句**

**executeUpdate() insert插入 update更新 delete删除**

**executeQuery() select查询**

**(5) 、处理结果集**

**(6) 、释放资源**

**jdbc完成对数据库的增删改查**

**增删改是：statement.executeUpdate 查询是：statement.executeQuery；**

**jdbc工具类的封装**

\* 此类专门封装jdbc操作

\* 封装获取数据库连接Connection的方法

\* 封装获取语句平台Statement的方法

\* 封装资源释放的方法

\*

\* 工具类封装的思路？？？

\* 1、工具类中的方法为了方便调用，都是类方法

\* 2、方法中如果出现了异常，只能try catch捕获，不能抛出

\* 3、工具类中的方法应该互相独立、互不干扰。高内聚、低耦合

**jdbc中的预编译语句平台**

有俩平台：

**Statement(普通的)：会出现SQL注入的情况**

**PreparedStatement(预编译平台)：可以有效的防止注入**

**why？**

**1、在创建语句平台时，已经指定了sql语句，在真正执行sql语句之前，已经对sql进行了预编译，确定了执行的功能以及语句中关键字的个数**！

**2、通过？绑定数据时，本质上就是纯粹的值替换，哪怕传入的值中有sql的关键字，也不识别、不生效**

**预编译平台与普通平台的区别是什么？？？**

**1、预编译平台在创建时需要指定sql语句，不确定的值以？替代，从而达到了预编译的效果，有效的防止了sql注入，在执行语句之前需要先通过setObject()来为？绑定值。而普通平台Statement在创建时没有指定sql语句，所以无法防止sql注入**

**2、创建语句平台的方法不一样：**

**普通平台：连接对象.createStatement()**

**预编译平台：连接对象.prepareStatement(sql语句)**

**3、执行语句时方法的参数不一样，但是方法名一致。增删改executeUpdate() 查询executeQuery()**

**普通平台：方法有参数，为要执行的sql语句**

**预编译平台：方法是无参的**

**4、预编译平台执行sql的效率高！因为预编译平台在执行sql是只需要执行即可，而普通平台需要先编译再执行**

关于预编译：

**获取预编译语句平台**

**参数直接传入要执行的sql语句，不确定的值使用？代替**

connection.prepareStatement("select \* from user where uname = ? and pwd = ?");

**如果sql语句中使用？来替代了值，那么在真正执行之前，需要为？绑定数据、绑定参数**

**预编译语句平台对象.setObject(？在sql语句中的位置, 为？绑定的值) ？的位置从1开始！！！**

pStat.setObject(1, uname);//1是位置，

//因为预编译平台在创建时已经指定了sql语句，那么执行方法时，就不需要再给定sql了，无参方法

set = pStat.executeQuery();

PreparedStatement 继承 Statement 是父类的引用指向子类的对象，一种多态的形式