

## 一、数据库驱动的概念、JDBC

数据库厂商提供的用来操作数据库用的jar包就是数据库驱动。各个厂商如果提供各自的数据库驱动的话会导致开发人员学习成本太高，所以sun公司提供了一套数据库驱动应该遵循的接口规范，这套规范就叫做JDBC，本质上是很多的接口。

由于所有的数据库驱动都遵循JDBC规范，是JDBC接口的一个实现，我们在学习和使用数据库时只要学习JDBC中的接口就可以了。

## 二、JDBC快速入门

\*在数据库中建立好表

\*在程序中导入数据库驱动包

### 1. 注册数据库驱动

DriverManager.registerDriver(new Driver()); //缺点一：观察mysqlDriver源码发现此方法导致了数据库驱动被注册了两次。缺点二：整个程序域mysql数据库驱动绑定增加了耦合性

```
Class.forName("com.mysql.jdbc.Driver");
```

### 2. 获取连接

```
DriverManager.getConnection(url, user, password);
```

~url的写法：

Oracle写法：jdbc:oracle:thin:@localhost:1521:sid

SqlServer—jdbc:microsoft:sqlserver://localhost:1433; DatabaseName=sid

MySQL—jdbc:mysql://localhost:3306/sid

~url可以接的参数

user、password

useUnicode=true&characterEncoding=UTF-8

### 3. 获取传输器

createStatement()：创建向数据库发送sql的statement对象。

prepareStatement(sql)：创建向数据库发送预编译sql的PreparedStatement对象。

### 4. 利用传输器执行sql语句获取结果集

executeQuery(String sql)：用于向数据库发送查询语句。

executeUpdate(String sql)：用于向数据库发送insert、update或delete语句

execute(String sql)：用于向数据库发送任意sql语句

### 5. 遍历结果集取出结构

ResultSet以表的样式在内存中保存了查询结果，其中还维护了一个游标，最开始的时候游标在第一行之前，每调用一次next()方法就试图下移一行，如果移动成功返回true；

ResultSet还提供了很多个Get方法，用来获取查询结果中的不同类型的数据

除了next方法，还有以下方法可以用来遍历结果集：

next()：移动到下一行

Previous()：移动到前一行

absolute(int row)：移动到指定行

beforeFirst()：移动到resultSet的最前面。

afterLast()：移动到resultSet的最后面。

## 6. 释放资源

conn是一个有限的资源，用完立即要释放  
stat占用内存，所以使用完后也要释放  
rs占用内存，所以使用完后也要释放  
释放时后创建的先释放

```
if(rs != null){  
    try {  
        rs.close();  
    } catch (SQLException e) {  
        e.printStackTrace();  
    } finally{  
        rs = null;  
    }  
}  
  
if(stat != null){  
    try {  
        stat.close();  
    } catch (SQLException e) {  
        e.printStackTrace();  
    } finally{  
        stat = null;  
    }  
}  
  
if(conn != null){  
    try {  
        conn.close();  
    } catch (SQLException e) {  
        e.printStackTrace();  
    } finally{  
        conn = null;  
    }  
}
```