回顾:

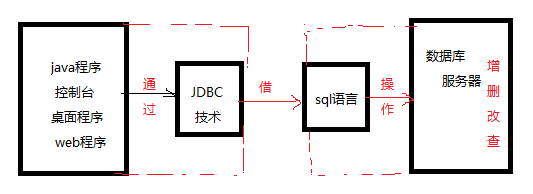
Java语言、web前端技术(html5、css3、Javascript、Jquery)、mysql数据库

JDBC

1. 什么是Jdbc:

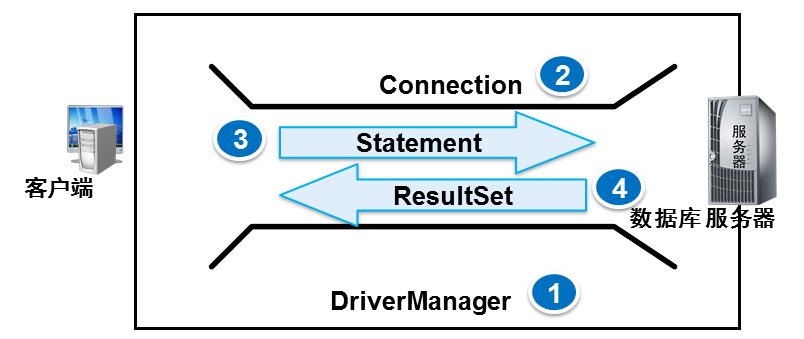
Jdbc是Java数据库连接技术的简称，提供连接各种常用数据库的能力.

注意:jdbc提供操作所有数据的统一标准



1. JDBC常用API(位于java.sql包中或javax.sql包)

Jdbc API主要功能;与数据建立连接，执行sql语句，处理结果.



* **DriverManager ：依据数据库的不同，管理JDBC驱动**
* **Connection ：负责连接数据库并担任传送数据的任务**
* **Statement ：由 Connection 产生、负责执行SQL语句**
* **ResultSet：负责保存Statement执行后所产生的查询结果**

三、jdbc操作数据库的步骤

|  |
| --- |
| *//准备工作:在工程中导入jdbc相应的驱动包(mysql-connector-java)  //将驱动jar包复制到工程中，并添加到工程库* **try** {  *//1.加载驱动类* Class.*forName*(**"com.mysql.jdbc.Driver"**);  *//2.使用DrvierManager类获取连接对象  //Connection con=DriverManager.getConnection(连接字符串,用户名,密码);  //连接字符串: jdbc:mysql://localhost:3306/数据库名称* Connection con=DriverManager.*getConnection*(**"jdbc:mysql://localhost:3306/k9051db?useUnicode=true&characterEncoding=utf-8"**,**"root"**,**"123"**);  *//3.创建statement对象* Statement st=con.createStatement();  *//4.执行sql  //4.1使用executeUpdate方法执行添加、修改、删除返回影响行数* **int** temp=st.executeUpdate(sql);  //4.2*使用executeQuery方法执行查询返回结果集(ResultSet)对象*  //ResultSet rs=st.executeQuery(sql);  *//5.关闭资源* st.close();  con.close();  st=**null**;  con=**null**;  **if**(temp>0)  System.***out***.println(**"添加成功"**);  **else** System.***out***.println(**"添加失败"**);  } **catch** (ClassNotFoundException e) {  e.printStackTrace();  } **catch** (SQLException e) {  e.printStackTrace();  } |

1. 结果集对象ResultSet
2. 什么是结果集:

结果集对象是指向查询结果的指针.默认停在第一行上面

1. 结果集对象常用方法

|  |
| --- |
| *a.使用next()将指针下移一行,如果有记录返回true,无记录返回false //rs.next(); b.使用getXXX(列号|列名)获取某列的值 列索引从1开始 //System.out.println("编号:"+rs.getInt(1)); //System.out.println("名称:"+rs.getString(2)); c.循环所有记录* System.***out***.println(**"编号\t名称"**); **while**(rs.next()){  System.***out***.println(rs.getObject(**"gid"**)+**"\t"**+rs.getObject(**"gname"**)); } |

示例:执行查询

|  |
| --- |
| *//查询年级* **public static void** main(String[] args) {  String sql=**"select** *\** **from grade where gid>3 order by gid desc"**;   *//执行sql* **try** {  Class.*forName*(**"com.mysql.jdbc.Driver"**);  Connection con=DriverManager.*getConnection*(**"jdbc:mysql://localhost:3306/k9051db?useUnicode=true&characterEncoding=utf-8"**,**"root"**,**"123"**);  Statement st=con.createStatement();  *//4.2使用executeQuery方法执行查询返回结果集(ResultSet)对象* ResultSet rs=st.executeQuery(sql);    *//a.使用next()将指针下移一行,如果有记录返回true,无记录返回false  //rs.next();  //b.使用getXXX(列号|列名)获取某列的值 列索引从1开始  //System.out.println("编号:"+rs.getInt(1));  //System.out.println("名称:"+rs.getString(2));  //c.循环所有记录* System.***out***.println(**"编号\t名称"**);  **while**(rs.next()){  System.***out***.println(rs.getObject(**"gid"**)+**"\t"**+rs.getObject(**"gname"**));  }   *//5.关闭资源* rs.close();  st.close();  con.close();  rs=**null**;  st=**null**;  con=**null**;  } **catch** (ClassNotFoundException e) {  e.printStackTrace();  } **catch** (SQLException e) {  e.printStackTrace();  }   } |