**本 科 实 验 报 告**

实验名称： 数据库第五次实验

|  |  |  |  |
| --- | --- | --- | --- |
| 学 员： | 于鹄杰 | 学 号： | 201506021075 |
| 培养类型： | 工程技术类 | 年 级： | 2015级 |
| 专 业： | 网络工程（军事物联网方向） | 所属学院： | 计算机学院 |
| 指导教员： | 周竞文 | 职 称： | 讲师 |

国防科技大学教务处制

**数据库编程**

## 1. 功能说明

利用JDBC数据库编程方法实现一个简单的学生信息管理系统，对实验2中建立的表S(Sno, Sname, Ssex, Sage)进行操作，系统应具有以下功能：

1）（**必选**）**插入**一个学生的信息；

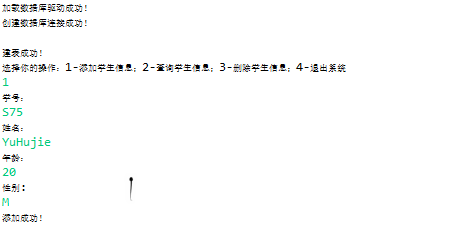
2）（**必选**）根据学号**查询**一个学生的信息；

3）（**必选**）根据学号**删除**一个学生的信息；

## 2. 测试用例及结果：

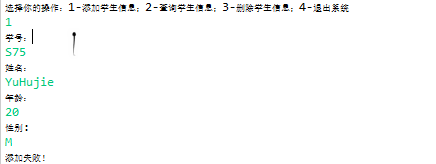
1）添加本人信息（**必须使用本人真实姓名的全拼**），要能正确添加；

截图结果如下：



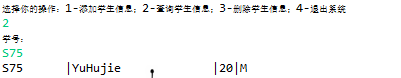
1. 再次添加本人信息（与上一步添加的信息相同），提示添加失败；

截图结果如下：



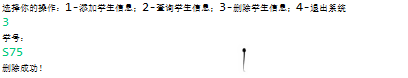
1. 根据学号查询本人信息，要能正确输出本人信息；

截图结果如下：



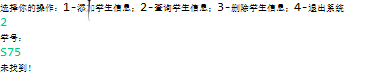
1. 根据学号删除本人信息，要能成功删除；

截图结果如下：

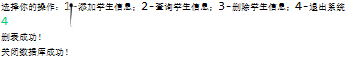


1. 再次查询本人信息，提示未找到；

截图结果如下：



6）退出系统；



## 3. 源代码：

**import** java.sql.Connection;

**import** java.sql.DriverManager;

**import** java.sql.SQLException;

**import** java.sql.PreparedStatement;

**import** java.sql.ResultSet;

**import** java.util.Scanner;

**public** **class** JDBCTest {

**private** Connection conn=**null**; //与Oracle的连接

**private** String JDBC\_DRIVER = "oracle.jdbc.driver.OracleDriver";//oracle驱动名

**private** String USER = "system"; //用户名

**private** String PWD = "123456"; //密码

**private** String DB\_URL = "jdbc:oracle:thin:@localhost:1521:xe";//Oracle服务地址：localhost为Oracle所在主机，1521为端口号，后面字段为数据库实例名（企业版为orcl，精简版为xe）

/\*主函数\*/

**public** **static** **void** main(String[] args){

JDBCTest c = **new** JDBCTest();

**short** option;

**boolean** ex=**false**;

Scanner dt = **new** Scanner(System.***in***);

String s\_no;

String s\_name;

String s\_sex;

**short** s\_age;

c.connectOracle();//在进行数据库操作之前要连接数据库

c.createStudentTable();

**while**(!ex)

{

System.***out***.println("选择你的操作：1-添加学生信息；2-查询学生信息；3-删除学生信息；4-退出系统");

option = dt.nextShort();

**switch**(option)

{

**case** 1:

System.***out***.println("学号：");

s\_no = dt.nextLine();

s\_no = dt.nextLine();

System.***out***.println("姓名：");

s\_name = dt.nextLine();

System.***out***.println("年龄：");

s\_age = dt.nextShort();

System.***out***.println("性别:");

s\_sex = dt.nextLine();

s\_sex = dt.nextLine();

c.insertStudent(s\_no, s\_name, s\_age, s\_sex);

**break**;

**case** 2:

System.***out***.println("学号：");

s\_no = dt.nextLine();

s\_no = dt.nextLine();

c.selectStudent(s\_no);

**break**;

**case** 3:

System.***out***.println("学号：");

s\_no = dt.nextLine();

s\_no = dt.nextLine();

c.deleteStudent(s\_no);

**break**;

**case** 4:

ex = **true**;

**break**;

**default**:

System.***out***.println("输入错误");

}

}

/\*此处可进行各种数据库操作，如建表、增删改查等\*/

c.dropStudentTable();

c.closeOracle();//操作完毕后要关闭数据库连接，释放资源

}

**public** Connection connectOracle(){

**try**{

Class.*forName*(JDBC\_DRIVER);//加载驱动

System.***out***.println("加载数据库驱动成功！");

}**catch**(ClassNotFoundException e){

System.***out***.println("加载数据库驱动失败！");

e.printStackTrace();

}

**try**{

conn = DriverManager.*getConnection*(DB\_URL, USER, PWD);//连接数据库

System.***out***.println("创建数据库连接成功！\n");

}**catch**(SQLException e){

System.***out***.print(conn);

System.***out***.println("创建数据库连接失败！");

conn = **null**;

e.printStackTrace();

}

**return** conn;

}

**public** **void** closeOracle(){

**try**{

**if**(conn != **null**){

conn.close();//关闭数据库

System.***out***.println("关闭数据库成功！");

}

}**catch**(SQLException e){

System.***out***.println("关闭数据库失败！");

e.printStackTrace();

}

}

**public** **void** createStudentTable(){

**try**{

String sql = "CREATE TABLE Student(Sno CHAR(9) PRIMARY KEY,Sname CHAR(20) NOT NULL,Sage NUMBER(3),Ssex CHAR(2) CHECK(Ssex IN('M','F')))";//SQL建表语句

PreparedStatement stmt = conn.prepareStatement(sql);//准备语句

stmt.execute();//执行语句

System.***out***.println("建表成功！");

}**catch**(SQLException e){

System.***out***.println("建表失败！");

//e.printStackTrace();

}

}

**public** **void** dropStudentTable(){

**try**{

String sql = "DROP TABLE Student";//SQL建表语句

PreparedStatement stmt = conn.prepareStatement(sql);//准备语句

stmt.execute();//执行语句

System.***out***.println("删表成功！");

}**catch**(SQLException e){

System.***out***.println("删表失败！");

}

}

**public** **void** insertStudent(String s\_no,String s\_name,**short** s\_age,String s\_sex){

**try**{

String sql = "INSERT INTO Student VALUES(?,?,?,?)";//SQL语句,问号为占位符，其值在后面设置

PreparedStatement stmt = conn.prepareStatement(sql);//准备语句

stmt.setString(1,s\_no);//为语句中的占位符设置具体值，setString是设置字符串值，将第1个占位符的值设置成"S1"

stmt.setString(2, s\_name);//将第2个占位符的值设置成

stmt.setShort(3, s\_age);

stmt.setString(4, s\_sex);

stmt.execute();//执行语句

System.***out***.println("添加成功！");

}**catch**(SQLException e){

System.***out***.println("添加失败！");

}

}

**public** **void** selectStudent(String s\_no){

**try**{

String sql = "SELECT \* FROM Student WHERE Sno="+"'"+s\_no+"'";//SQL语句

PreparedStatement stmt = conn.prepareStatement(sql);//准备语句

ResultSet rs=stmt.executeQuery();//执行查询语句，查询结果存放在变量rs中

**if**(!rs.next()) System.***out***.println("未找到！");

**else** rs=stmt.executeQuery();

**while**(rs.next()){//获取rs中的查询结果，类似迭代器的使用

String sno=rs.getString(1);//以字符串类型获取当前元组的第1个分量

String sname=rs.getString(2);//以字符串类型获取当前元组的第2个分量

**short** sage=rs.getShort(3);

String ssex = rs.getString(4);

System.***out***.println(sno+'|'+sname+'|'+sage+'|'+ssex);

}

}**catch**(SQLException e){

System.***out***.println("未找到！");

}

}

**public** **void** deleteStudent(String s\_no){

**try**{

String sql = "DELETE FROM Student WHERE Sno="+"'"+s\_no+"'";//SQL语句,问号为占位符，其值在后面设置

PreparedStatement stmt = conn.prepareStatement(sql);//准备语句

stmt.execute();//执行语句

System.***out***.println("删除成功！");

}**catch**(SQLException e){

System.***out***.println("删除失败！");

}

}

}