实验五 综合实验

实验日期： 年 月 日 班级：

学号（后四位）： 姓名： 成绩：

1. 实验目的

1 了解JDBC的作用，掌握通过JDBC访问数据库的方法；

2 掌握简单的sql语句；

3 掌握常用数据库操作的Java语句，对后台数据库进行增删改交互操作。

二．支撑的课程目标

本实验项目可以支撑“课程目标1：能根据软件工程领域的复杂工程问题，进行研究，运用Java的编程思想设计项目开发方案，并按照合理步骤和软件工程流程进行方案的实施”、“课程目标2：掌握Java语言的基本特点、语法和面向对象的编程思想，能应用Java语言的编程原理、方法和技术，针对软件工程领域中复杂工程问题的设计解决方案要求进行软件开发”。

本实验通过“任务驱动”方法，使学生根据软件工程中项目需求，按照面向对象的思想，设计项目的实施方案，运用Java的相关知识点，按照方案步骤进行满足任务需要的开发。

三．实验内容

使用建立一个Student类，利用数据库来存储多个Student，写完一个方法在main中写一段测试代码，运行以保证目前所做工作的正确性。有以下方法：

（1）add(Student stu)：可以向其中增加新的学生，并保存在数据库中。

测试add方法是否正确：用add方法向数据库增加一个新的学生，然后在数据库的图形管理界面中查询，确认是否增加。

（2）dispAll()：可以显示所有的学生信息。

（3）findById(long id)：可以按照学号来查找，并显示符合条件的学生信息，查无该人的话显示错误信息。

（4）findByName(String name)：可以按照姓名查找学生，找到后显示其信息，查无此人显示错误信息。

（5）delById(long id)：可以按照id删除学生的信息，然后显示找到该人。若查无此人，显示相应的错误信息。

（6）sortByXXX：可以按照指定的字段排序，并显示排序后的信息。

**代码：**

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.sql.Statement;

import java.sql.Timestamp;

import util.JDBCutil;

public class Experiment5 {

private Connection conn = null;

public PreparedStatement ps = null;

public Experiment5(Connection con) {

this.conn = con;

}

public void add(Student stu) {

try {

String sql = "insert into stu (id,name,age,sex,logintime) values (?,?,?,?,?)";

ps = conn.prepareStatement(sql);

ps.setString(1, stu.getID());

ps.setString(2, stu.getName());

ps.setInt(3, stu.getAge());

ps.setObject(4, stu.getSex()+"");

Timestamp timestamp = new Timestamp(System.currentTimeMillis());

ps.setTimestamp(5,timestamp);

ps.execute();

} catch (SQLException e) {

e.printStackTrace();

}finally {

JDBCutil.close(ps);

}

}

public void dispAll() {

ResultSet rs = null;

String sql = "select \* from stu where 1=1";

try {

ps = conn.prepareStatement(sql);

rs = ps.executeQuery();

while(rs.next()) {

System.out.println("id:"+rs.getString(1) + " " +"name:"+ rs.getString(2) + " " +"age:"+ rs.getInt(3) + " " +

"sex:" + rs.getString(4) + " " +"logintime:"+ rs.getTimestamp(5));

}

ps.execute();

} catch (SQLException e) {

e.printStackTrace();

}finally {

JDBCutil.close(ps);

JDBCutil.close(rs);

}

}

public boolean findById(String id) {

ResultSet rs = null;

String sql = "select \* from stu where id=?";

try {

ps = conn.prepareStatement(sql);

ps.setString(1, id);

rs = ps.executeQuery();

while(rs.next()) {

System.out.println("id查询结果：");

System.out.println("id:"+rs.getString(1) + " " +"name:"+ rs.getString(2) + " " +"age:"+ rs.getInt(3) + " " +

"sex:" + rs.getString(4) + " " +"logintime:"+ rs.getTimestamp(5));

return true;

}

} catch (SQLException e) {

e.printStackTrace();

}finally {

JDBCutil.close(ps);

JDBCutil.close(rs);

}

return false;

}

public boolean findByName(String name) {

ResultSet rs = null;

String sql = "select \* from stu where name=?";

try {

ps = conn.prepareStatement(sql);

ps.setString(1, name);

rs = ps.executeQuery();

while(rs.next()) {

System.out.println("name查询结果：");

System.out.println("id:"+rs.getString(1) + " " +"name:"+ rs.getString(2) + " " +"age:"+ rs.getInt(3) + " " +

"sex:" + rs.getString(4) + " " +"logintime:"+ rs.getTimestamp(5));

return true;

}

} catch (SQLException e) {

e.printStackTrace();

}finally {

JDBCutil.close(ps);

JDBCutil.close(rs);

}

return false;

}

public int delById(String id) {

Statement st = null;

String sql = "delete from stu where id=" + id;

try {

st = conn.prepareStatement(sql);

return st.executeUpdate(sql);

} catch (SQLException e) {

e.printStackTrace();

}

return 0;

}

}

import java.sql.Connection;

import util.JDBCutil;

public class Experiment5Test {

public static void main(String[] args) {

Connection conn = JDBCutil.getMysqlConn();

Experiment5 operasql = new Experiment5(conn);

for (int i = 0; i <50; i++) {

Student stu = new Student("zhangsan"+i, 18, 'm', "2017160401"+i);

operasql.add(stu);

}

/\*\* 显示信息 \*/

operasql.dispAll();

/\*\* 按照id查找 \*/

if(operasql.findById("201716040111")) {

}else {

System.out.println("未找到！");

}

/\*\* 按照姓名查找 \*/

if(operasql.findByName("zhangsan1")) {

}else {

System.out.println("未找到！");

}

/\*\* 按照id删除 \*/

int count = operasql.delById("201716040114");

if(count == 0) {

System.out.println("删除失败！");

}else if(count == 1) {

System.out.println("删除成功！");

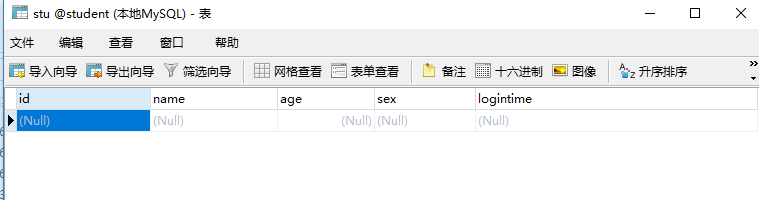
}

}

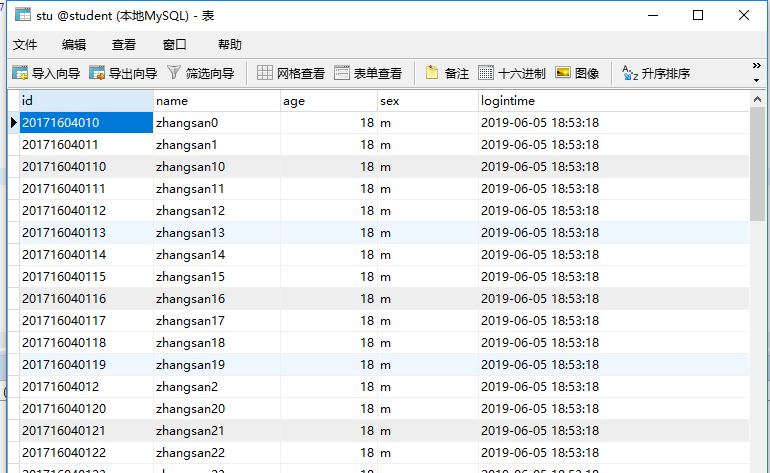
}

**实验截图：**

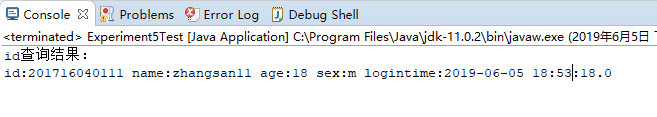
**添加前**



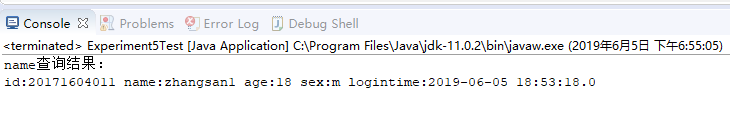
**添加后**



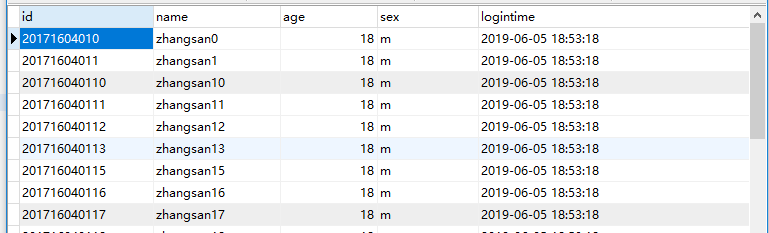
**按照学号查找**



**按照姓名查找**



**按照id删除14**



四．在本次实验中的问题和解决方法

五．心得体会

注意数据库的语法以及连接的数据库名字，数据库表中一定要定义主键