实验四 数据库编程（2 学时）

1、实验目的：

（1）掌握基本的DBMS的使用

（2）理解JDBC的概念

（3）运用JDBC-ODBC桥开发数据库系统

2、基本要求：

（1）注意自始至终贯彻课程中所介绍程序设计风格，养成良好的编程习惯。

（2）独立完成所布置习题。

（3）为保证尽量在统一安排的上机时间内编译运行通过程序，应事先设计好程序。

（4）认真完成每次实验，并写出实验报告。

3、实验内容与步骤：

完成一个具体的数据库系统的开发，如图书管理系统、人事管理系统、学生信息管理系统等等。

**SQLMethod**

**package** mySQL;  
  
**public interface** SQLMethod {  
 **void** insertData(); *//增* **void** deleteData(); *//删* **void** updateDate(); *//改* **void** selectAll(); *//查*}

**DBHelper.java**

**package** mySQL;  
  
**import** java.sql.\*;  
  
**public class** DBHelper {  
 **private static** Connection *con* = **null**;  
 *//数据库的url地址；?useUnicode=true&characterEncoding=utf-8解决数据库中文乱码问题* **private static** String *url* = **"jdbc:mysql://localhost:3306/student?&useSSL=true&characterEncoding=utf-8&serverTimezone=GMT"**;  
 *//驱动的完整包名* **private static** String *driver* = **"com.mysql.cj.jdbc.Driver"**;  
 *//数据库的用户名* **private static** String *user* = **"root"**;  
 *//数据库的用户名密码* **private static** String *password* = **"212033"**;  
  
 **static**{  
 **try**{  
 Class.*forName*(*driver*); *//加载JDBC-MySQL驱动* }**catch** (Exception e){  
 e.printStackTrace();  
 }  
 **try** {  
 *con* = DriverManager.*getConnection*(*url*, *user*, *password*);  
 } **catch** (Exception e) {  
 e.printStackTrace();  
 }  
 }  
  
 *//构造方法私有化，安全* **private** DBHelper(){  
  
 }  
  
 *//只对外提供这一个方法，确保单例，同时封装性更好* **public static** Connection getConnect(){  
 **return** *con*;  
 }  
  
 *//关闭连接* **public static void** closeConnect(){  
 **try** {  
 *con*.close();  
 } **catch** (SQLException e) {  
 e.printStackTrace();  
 }  
 }  
  
 *//查询数据库中指定表* **public static void** SQLExecute(Connection connection){  
  
 }  
}

**StudentTable.java**

**package** mySQL;  
  
**import** java.sql.\*;  
**import** java.util.Scanner;  
  
**public class** StudentTable **implements** SQLMethod {  
  
 @Override  
 **public void** selectAll() {  
 Connection connection = **null**;  
 Statement sql = **null**;  
 ResultSet resultSet = **null**;  
 **try** {  
 connection = DBHelper.*getConnect*();  
 sql = connection.createStatement();  
 resultSet = sql.executeQuery(**"SELECT \* FROM student"**); *//查询student表* **while** (resultSet.next()){  
 String number = resultSet.getString(1);  
 String name = resultSet.getString(2);  
 String sex = resultSet.getString(3);  
 **int** age = resultSet.getInt(4);  
 String dept = resultSet.getString(5);  
 System.***out***.printf(**"%s\t"**,number);  
 System.***out***.printf(**"%s\t"**,name);  
 System.***out***.printf(**"%s\t"**,sex);  
 System.***out***.printf(**"%d\t"**,age);  
 System.***out***.printf(**"%s\n"**,dept);  
 }  
 }**catch** (SQLException e){  
 System.***out***.println(e);  
 }  
 }  
  
 @Override  
 **public void** insertData() {  
 Connection connection = **null**;  
 PreparedStatement statement = **null**;  
 Scanner scanner = **new** Scanner(System.***in***);  
 *//构造交互式输入* System.***out***.println(**"请输入想要插入的数据（学号、姓名、性别、年龄、院系）："**);  
 String number = scanner.next();  
 String name = scanner.next();  
 String sex = scanner.next();  
 **int** age = scanner.nextInt();  
 String dept = scanner.next();  
 *//使用动态参数构造SQL语句* String sqlStr = **"INSERT INTO student VALUES(?,?,?,?,?)"**;  
 **try** {  
 connection = DBHelper.*getConnect*();  
 statement = connection.prepareStatement(sqlStr);  
 statement.setString(1,number); *//设置参数数值* statement.setString(2,name); *//设置参数数值* statement.setString(3,sex); *//设置参数数值* statement.setInt(4,age); *//设置参数数值* statement.setString(5,dept); *//设置参数数值* statement.executeUpdate(); *//执行构造的SQL语句* } **catch** (SQLException e) {  
 e.printStackTrace();  
 }  
 System.***out***.println(**"插入新学生后的student表："**);  
 selectAll();  
 }  
  
 @Override  
 **public void** deleteData() {  
 Connection connection = **null**;  
 PreparedStatement statement = **null**;  
 *//构造交互式输入* Scanner scanner = **new** Scanner(System.***in***);  
 System.***out***.println(**"请输想要删除的学生学号："**);  
 String name = scanner.next();  
 *//使用动态参数构造SQL语句* String sqlDelete = **"DELETE FROM student WHERE Sno = ? "**;  
 **try** {  
 connection = DBHelper.*getConnect*();  
 statement = connection.prepareStatement(sqlDelete);  
 statement.setString(1,name); *//设置参数数值* statement.executeUpdate();  
 } **catch** (SQLException e) {  
 e.printStackTrace();  
 }  
 System.***out***.println(**"删除成功后的student表："**);  
 selectAll();  
 }  
  
 @Override  
 **public void** updateDate() {  
 Connection connection = **null**;  
 PreparedStatement statement = **null**;  
 *//构造交互式输入* Scanner scanner = **new** Scanner(System.***in***);  
 System.***out***.println(**"输入想要更新的学生学号："**);  
 String number = scanner.next();  
 System.***out***.println(**"请输入更新后的年龄："**);  
 **int** age = scanner.nextInt();  
 *//设置动态参数(?)构造SQL语句* String sqlChange = **"UPDATE student SET Sage = ? WHERE Sno = ? "**;  
 **try** {  
 connection = DBHelper.*getConnect*();  
 statement = connection.prepareStatement(sqlChange);  
 statement.setInt(1,age); *//设置参数数值* statement.setString(2,number); *//设置参数数值* System.***out***.println(sqlChange);  
 statement.executeUpdate();  
 } **catch** (SQLException e) {  
 e.printStackTrace();  
 }  
 System.***out***.println(**"更新成功后的student表："**);  
 selectAll();  
 }  
}

**mySQL.java**

**package** mySQL;  
  
**import** java.sql.\*;  
**import** java.util.Scanner;  
  
**public class** mySQL {  
 **public static void** main(String[] args) {  
 System.***out***.println(**"---------学生信息管理系统---------"**);  
 System.***out***.println(**"| 1.输出所有学会信息 |"**);  
 System.***out***.println(**"| 2.添加学生信息 |"**);  
 System.***out***.println(**"| 3.删除对应学生信息 |"**);  
 System.***out***.println(**"| 4.更改对应学生信息 |"**);  
 System.***out***.println(**"--------------------------------"**);  
 Scanner scanner = **new** Scanner(System.***in***);  
 System.***out***.println(**"请输入操作："**);  
 String input = scanner.next();  
 StudentTable table = **new** StudentTable();  
 **while** (scanner.hasNextLine() && !input.equals(**"0"**)){  
 **switch** (input){  
 **case "1"**:  
 System.***out***.println(**"---------所有学会信息---------"**);  
 table.selectAll();  
 **break**;  
 **case "2"**:  
 table.insertData();  
 **break**;  
 **case "3"**:  
 table.deleteData();  
 **break**;  
 **case "4"**:  
 table.updateDate();  
 **break**;  
 }  
 System.***out***.println(**"请输入操作："**);  
 input = scanner.next();  
 }  
 DBHelper.*closeConnect*();  
 }  
}

 