**package** assignment3;

**import** java.io.Serializable;

**public** **class** Student **implements** Serializable{

**public** **int** rollNo;

**public** String name;

**public** **double** marks;

**public** Student(**int** rollNo, String name, **double** marks) {

**this**.rollNo = rollNo;

**this**.name = name;

**this**.marks = marks;

}

@Override

**public** String toString() {

**return** "Student [rollNo=" + rollNo + ", name=" + name + ", marks="

+ marks + "]";

}

}

package assignment3;

import java.sql.Connection;

import java.sql.PreparedStatement;

import java.sql.ResultSet;

import java.sql.SQLException;

import java.util.ArrayList;

public class Operations {

int addStudent(Student st) {

int rowsInserted = 0;

Connection con = DB.getConnection();

try {

PreparedStatement ps = con.prepareStatement("insert into Student(rollNo, name, marks) values(?,?,?)");

ps.setInt(1, st.rollNo);

ps.setString(2, st.name);

ps.setDouble(3, st.marks);

rowsInserted = ps.executeUpdate();

con.close();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

return rowsInserted;

}

int deleteStudent(int rollNo) {

int rowsDeleted = 0;

Connection con = DB.getConnection();

try {

PreparedStatement ps = con.prepareStatement("delete from Student where rollNo = ?");

ps.setInt(1, rollNo);

rowsDeleted = ps.executeUpdate();

con.close();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

return rowsDeleted;

}

Student searchStudent(int rollNo){

Student st = null;

Connection con = DB.getConnection();

try {

PreparedStatement ps = con.prepareStatement("Select \* from Student where rollNo = ?");

ps.setInt(1, rollNo);

ResultSet rs = ps.executeQuery();

if(rs.next()){

st = new Student(rs.getInt("rollNo"), rs.getString("name"), rs.getDouble("marks"));

}

con.close();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

return st;

}

ArrayList<Student> failedStudents(double pm){

ArrayList<Student> failed = new ArrayList<Student>();

Connection con = DB.getConnection();

try {

PreparedStatement ps = con.prepareStatement("Select \* from Student where marks < ?");

ps.setDouble(1, pm);

ResultSet rs = ps.executeQuery();

while(rs.next()){

Student st = new Student(rs.getInt("rollNo"), rs.getString("name"), rs.getDouble("marks"));

failed.add(st);

}

con.close();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

return failed;

}

ArrayList<Student> display(){

ArrayList<Student> allData = new ArrayList<Student>();

Connection con = DB.getConnection();

try {

PreparedStatement ps = con.prepareStatement("Select \* from Student");

ResultSet rs = ps.executeQuery();

while(rs.next()){

Student st = new Student(rs.getInt("rollNo"), rs.getString("name"), rs.getDouble("marks"));

allData.add(st);

}

con.close();

} catch (SQLException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

return allData;

}

}

**package** assignment3;

**import** java.sql.Connection;

**import** java.sql.DriverManager;

**import** java.sql.SQLException;

**public** **class** DB {

**static** Connection *con* = **null**;

**public** **static** Connection getConnection() {

**try** {

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

*con* = DriverManager.*getConnection*(

"jdbc:mysql://localhost:3306/studentdatabase", "root", "root");

} **catch** (SQLException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

} **catch** (ClassNotFoundException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

**return** *con*;

}

**public** **static** **void** closeConnection(){

**try** {

*con*.close();

} **catch** (SQLException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

}

}

package assignment3;

import java.io.DataInputStream;

import java.io.DataOutputStream;

import java.io.IOException;

import java.io.ObjectInputStream;

import java.io.ObjectOutputStream;

import java.net.Socket;

import java.util.Scanner;

public class Client {

public static void main(String[] args) {

// TODO Auto-generated method stub

Scanner sc= new Scanner(System.in);

try {

Socket cs= new Socket("localhost", 4000);

DataOutputStream dout = new DataOutputStream(cs.getOutputStream());

DataInputStream din = new DataInputStream(cs.getInputStream());

ObjectOutputStream oout = new ObjectOutputStream(

cs.getOutputStream());

ObjectInputStream oin = new ObjectInputStream(cs.getInputStream());

int ch;

String name;

int rollNo;

double marks;

// Student class of Client

Student st = null;

while (true) {

System.out.print("\n\*\*\*\*\* Select Operation \*\*\*\*\*\n");

System.out.print("\n\t1. Add Student\n\t2. Delete Student\n\t3. Search Student\n\t4. Failed Student\n\t5. Exit");

System.out.print("\nEnter choice: ");

ch = sc.nextInt();

dout.writeInt(ch);

switch (ch) {

case 1:

System.out.print("\nEnter Roll No - ");

rollNo = sc.nextInt();

System.out.print("\nEnter Name - ");

name = sc.next();

System.out.print("\nEnter Marks - ");

marks = sc.nextDouble();

st = new Student(rollNo, name, marks);

oout.writeObject(st);

System.out.print("\nAdded Succesfully " + rollNo

+ "\nCurrent Students ->" + din.readUTF());

break;

case 2:

System.out.print("\nEnter rollNo to be deleted : ");

rollNo = sc.nextInt();

dout.writeInt(rollNo);

System.out.print("\nDeleted Succesfully " + rollNo

+ "\nCurrent Students List" + din.readUTF());

break;

case 3:

System.out.print("\nEnter rollNo to be searched : ");

rollNo = sc.nextInt();

dout.writeInt(rollNo);

Student s = (Student) oin.readObject();

if (s != null) {

System.out.print("\nSearched Student -> " + s);

} else {

System.out.println("User not found !!!");

}

break;

case 4:

System.out.print("\nEnter passing marks : ");

double pm = sc.nextDouble();

dout.writeDouble(pm);

System.out.print("\nFailed List !!!\n" + din.readUTF());

break;

}

if (ch == 5) {

System.out.println(din.readUTF());

break;

}

}

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

} catch (ClassNotFoundException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

package assignment3;

import java.io.DataInputStream;

import java.io.DataOutputStream;

import java.io.IOException;

import java.io.ObjectInputStream;

import java.io.ObjectOutputStream;

import java.net.ServerSocket;

import java.net.Socket;

import java.util.ArrayList;

public class ServerDemo {

static ArrayList<Student> al = new ArrayList<Student>();

public static void main(String[] args) {

// TODO Auto-generated method stub

try {

ServerSocket ss = new ServerSocket(4000);

System.out.println("Server waiting for client....");

Socket cs = ss.accept();

DataOutputStream dout = new DataOutputStream(cs.getOutputStream());

DataInputStream din = new DataInputStream(cs.getInputStream());

ObjectOutputStream oout = new ObjectOutputStream(

cs.getOutputStream());

ObjectInputStream oin = new ObjectInputStream(cs.getInputStream());

Student st = null;

int rollNo;

Operations obj = new Operations();

while (true) {

int ch = din.readInt();

if (ch == 1) {

// Reading com.server.Student Object from client

st = (Student) oin.readObject();

// Calling addStudent() of com.server.Operations class

obj.addStudent(st);

dout.writeUTF(al.toString());

} else if (ch == 2) {

// Reading rollNo from client for deleting

rollNo = din.readInt();

// Calling addStudent() of com.server.Operations class

obj.deleteStudent(rollNo);

dout.writeUTF(al.toString());

} else if (ch == 3) {

// Reading rollNo from client for deleting

rollNo = din.readInt();

// Calling addStudent() of com.server.Operations class

st = obj.searchStudent(rollNo);

if (st != null) {

// Writing searched Student object back to client

oout.writeObject(st);

} else {

// Writing ArrayList al object back to client

oout.writeObject(null);

}

} else if (ch == 4) {

// Reading rollNo from client for deleting

double pm = din.readDouble();

// Calling addStudent() of com.server.Operations class

ArrayList<Student> failed = obj.failedStudents(pm);

dout.writeUTF(failed.toString());

} else if (ch == 5) {

dout.writeUTF("Bye Bye Client from server!!! ");

System.out.println("Socket Closed !!!s");

cs.close();

break;

}

}

ss.close();

} catch (IOException e) {

// TODO Auto-generated catch block

e.printStackTrace();

} catch (ClassNotFoundException e) {

// TODO Auto-generated catch block

e.printStackTrace();

}

}

}

OUTPUT-

\*\*\*\*\* Select Operation \*\*\*\*\*

1. Add Student

2. Delete Student

3. Search Student

4. Failed Student

5. Exit

Enter choice: 1

Enter Roll No - 18

Enter Name - dee

Enter Marks - 65

Added Succesfully 18

Current Students ->[]

\*\*\*\*\* Select Operation \*\*\*\*\*

1. Add Student

2. Delete Student

3. Search Student

4. Failed Student

5. Exit

Enter choice: 1

Enter Roll No - 88

Enter Name - sandip

Enter Marks - 90

Added Succesfully 88

Current Students ->[]

\*\*\*\*\* Select Operation \*\*\*\*\*

1. Add Student

2. Delete Student

3. Search Student

4. Failed Student

5. Exit

Enter choice: 2

Enter rollNo to be deleted : 18

Deleted Succesfully 18

Current Students List[]

\*\*\*\*\* Select Operation \*\*\*\*\*

1. Add Student

2. Delete Student

3. Search Student

4. Failed Student

5. Exit

Enter choice: 1

Enter Roll No - 20

Enter Name - shree

Enter Marks - 40

Added Succesfully 20

Current Students ->[]

\*\*\*\*\* Select Operation \*\*\*\*\*

1. Add Student

2. Delete Student

3. Search Student

4. Failed Student

5. Exit

Enter choice: 3

Enter rollNo to be searched : 20

Searched Student -> Student [rollNo=20, name=shree, marks=40.0]

\*\*\*\*\* Select Operation \*\*\*\*\*

1. Add Student

2. Delete Student

3. Search Student

4. Failed Student

5. Exit

Enter choice: 4

Enter passing marks : 45

Failed List !!!

[Student [rollNo=20, name=shree, marks=40.0]]

\*\*\*\*\* Select Operation \*\*\*\*\*

1. Add Student

2. Delete Student

3. Search Student

4. Failed Student

5. Exit

Enter choice: 5

Bye Bye Client from server!!!