JAVA LAB

WEEK 11

JDBC AND GUI

SABHARI P

2018103582

**TO DESIGN A GUI BASED JDBC DATABASE AND HELP TO MODIFY AND ACCESS DATA.**

**STUDENT.JAVA:**

package com.sabhari;

public class Student {

int RollNo;

String name;

int age;

String gender;

String address;

int mark1,mark2,mark3,mark4,mark5;

double CGPA;

public Student(int RollNo,String name,int age,String gender,String address,int mark1,int mark2,int mark3,int mark4,int mark5){

this.RollNo = RollNo;

this.name = name;

this.age = age;

this.gender = gender;

this.address = address;

this.mark1 = mark1;

this.mark2 = mark2;

this.mark3 = mark3;

this.mark4 = mark4;

this.mark5 = mark5;

this.CGPA = CGPACalculate(mark1,mark2,mark3,mark4,mark5);

}

public static double CGPACalculate(int mark1,int mark2,int mark3,int mark4,int mark5){

double CGPA = 0.0;

CGPA = (mark1\*5 + mark2\*5 + mark3\*5 + mark4\*5 + mark5\*5)/25;

CGPA = CGPA/10;

return CGPA;

}

}

**GUI.JAVA:**

package com.sabhari;

import javax.swing.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import java.sql.\*;

class Work{

public static void main(String[] args) {

GUI app = new GUI();

}

}

public class GUI implements ActionListener {

private Statement st = null;

JFrame frame1;

JLabel lab1,lab2,lab3,lab4,lab5,lab6,lab7,lab8,lab9,lab10,lab11,labres;

JPanel pan1,pan2,pan3,pan4,pan5,pan6,pan7,pan8,pan9,pan10,pan11,panres;

JTextField tf1,tf2,tf3,tf4,tf5,tf6,tf7,tf8,tf9,tf10;

JButton insert,update,delete,display,end;

private Connection con = null;

public GUI(){

String url = "jdbc:mysql://localhost/studentinfo?useUnicode=true&useJDBCCompliantTimezoneShift=true&useLegacyDatetimeCode=false&serverTimezone=UTC";

String UserName = "sabhari";

String PassWord = "2000";

try {

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

} catch (ClassNotFoundException e) {

e.printStackTrace();

System.out.println("Error in Registering jdbc");

}

try {

con = DriverManager.getConnection(url,UserName,PassWord);

System.out.println("Successfully connected to JDBC");

} catch (SQLException throwables) {

throwables.printStackTrace();

System.out.println("Error in connecting to jdbc");

}

frame1 = new JFrame("STUDENT DATABASE" );

lab1 = new JLabel("STUDENTS RECORDS");

pan1 = new JPanel();

pan1.add(lab1);

pan1.setBounds(380,10,200,20);

frame1.add(pan1);

lab2 = new JLabel("ROLL NO");

pan2 = new JPanel();

pan2.add(lab2);

pan2.setBounds(250,50,100,20);

frame1.add(pan2);

tf1 = new JTextField();

tf1.setBounds(500,50,250,20);

frame1.add(tf1);

lab3 = new JLabel("NAME");

pan3 = new JPanel();

pan3.add(lab3);

pan3.setBounds(250,90,100,20);

frame1.add(pan3);

tf2 = new JTextField();

tf2.setBounds(500,90,250,20);

frame1.add(tf2);

lab4 = new JLabel("AGE");

pan4 = new JPanel();

pan4.add(lab4);

pan4.setBounds(250,130,100,20);

frame1.add(pan4);

tf3 = new JTextField();

tf3.setBounds(500,130,250,20);

frame1.add(tf3);

lab5 = new JLabel("GENDER");

pan5 = new JPanel();

pan5.add(lab5);

pan5.setBounds(250,170,100,20);

frame1.add(pan5);

tf4 = new JTextField();

tf4.setBounds(500,170,250,20);

frame1.add(tf4);

lab6 = new JLabel("ADDRESS");

pan6 = new JPanel();

pan6.add(lab6);

pan6.setBounds(250,210,100,20);

frame1.add(pan6);

tf5 = new JTextField();

tf5.setBounds(500,210,250,20);

frame1.add(tf5);

lab7 = new JLabel("MARK 1");

pan7 = new JPanel();

pan7.add(lab7);

pan7.setBounds(250,250,100,20);

frame1.add(pan7);

tf6 = new JTextField();

tf6.setBounds(500,250,250,20);

frame1.add(tf6);

lab8 = new JLabel("MARK 2");

pan8 = new JPanel();

pan8.add(lab8);

pan8.setBounds(250,290,100,20);

frame1.add(pan8);

tf7 = new JTextField();

tf7.setBounds(500,290,250,20);

frame1.add(tf7);

lab9 = new JLabel("MARK 3");

pan9 = new JPanel();

pan9.add(lab9);

pan9.setBounds(250,330,100,20);

frame1.add(pan9);

tf8 = new JTextField();

tf8.setBounds(500,330,250,20);

frame1.add(tf8);

lab10 = new JLabel("MARK 4");

pan10 = new JPanel();

pan10.add(lab10);

pan10.setBounds(250,370,100,20);

frame1.add(pan10);

tf9 = new JTextField();

tf9.setBounds(500,370,250,20);

frame1.add(tf9);

lab11 = new JLabel("MARK 5");

pan11 = new JPanel();

pan11.add(lab11);

pan11.setBounds(250,410,100,20);

frame1.add(pan11);

tf10 = new JTextField();

tf10.setBounds(500,410,250,20);

frame1.add(tf10);

insert = new JButton("INSERT");

insert.setBounds(140,500,100,20);

frame1.add(insert);

insert.addActionListener(this::actionPerformed);

update = new JButton("UPDATE");

update.setBounds(300,500,100,20);

frame1.add(update);

update.addActionListener(this::actionPerformed);

delete = new JButton("DELETE");

delete.setBounds(460,500,100,20);

frame1.add(delete);

delete.addActionListener(this::actionPerformed);

display = new JButton("DISPLAY");

display.setBounds(600,500,100,20);

frame1.add(display);

display.addActionListener(this::actionPerformed);

end = new JButton("END");

end.setBounds(800,500,100,20);

frame1.add(end);

end.addActionListener(this::actionPerformed);

labres = new JLabel("The result will appear here!!");

panres = new JPanel();

panres.add(labres);

panres.setBounds(10,550,1000,80);

frame1.add(panres);

frame1.setSize(1000,700);

frame1.setDefaultCloseOperation(WindowConstants.EXIT\_ON\_CLOSE);

frame1.setLayout(null);

frame1.setVisible(true);

}

@Override

public void actionPerformed(ActionEvent e) {

if (e.getSource() == insert){

int RollNo = Integer.parseInt(tf1.getText());

String name = tf2.getText();

int age = Integer.parseInt(tf3.getText());

String gender = tf4.getText();

String address = tf5.getText();

int mark1 = Integer.parseInt(tf6.getText());

int mark2 = Integer.parseInt(tf7.getText());

int mark3 = Integer.parseInt(tf8.getText());

int mark4 = Integer.parseInt(tf9.getText());

int mark5 = Integer.parseInt(tf10.getText());

double CGPA = Student.CGPACalculate(mark1,mark2,mark3,mark4,mark5);

String query = "INSERT INTO `rollstudent`(`RollNo`, `name`, `age`, `gender`, `address`, `mark1`, `mark2`, `mark3`, `mark4`, `mark5`, `CGPA`) VALUES (\""+RollNo+"\",\"" + name + "\",\"" + age + "\",\"" + gender + "\",\"" + address + "\",\"" + mark1 + "\",\"" + mark2 + "\",\"" + mark3 + "\",\"" + mark4 + "\",\"" + mark5 + "\",\"" + CGPA +"\");";

Statement st = null;

try {

st = con.createStatement();

} catch (SQLException throwables) {

throwables.printStackTrace();

System.out.println("Error in statement");

}

int affected = 0;

try {

affected = st.executeUpdate(query);

} catch (SQLException throwables) {

throwables.printStackTrace();

System.out.println("Error in entering into table");

}

System.out.println(affected + "rows were affected");

labres.setText(affected + "rows were affected ");

tf1.setText("");

tf2.setText("");

tf3.setText("");

tf4.setText("");

tf5.setText("");

tf6.setText("");

tf7.setText("");

tf8.setText("");

tf9.setText("");

tf10.setText("");

}

if (e.getSource()==display) {

String query = "SELECT \* FROM `rollstudent`";

Statement st = null;

try {

st = con.createStatement();

} catch (SQLException throwables) {

throwables.printStackTrace();

System.out.println("Error in statement");

}

ResultSet rs = null;

try {

rs = st.executeQuery(query);

} catch (SQLException throwables) {

throwables.printStackTrace();

}

try {

while (rs.next()){

String record = "ROLL NO = " + rs.getInt(1) + " \tNAME = " + rs.getString(2) + " \tAGE = " + rs.getInt(3) + " \tGENDER = " + rs.getString(4) + " \tADDRESS = " + rs.getString(5) + " \tMARK 1 = " + rs.getInt(6) + " \tMARK 2 = " + rs.getInt(7) + " \tMARK 3 = " + rs.getInt(8) + " \tMARK 4 = " + rs.getInt(9) + " \tMARK 5 = " + rs.getInt(10) + " \tCGPA = " + rs.getDouble(11) +"";

labres.setText(record);

System.out.println(record);

}

}catch (Exception exc){

exc.printStackTrace();

}

}

if (e.getSource()==delete){

int RollNo = Integer.parseInt(tf1.getText());

System.out.println("Requested to delete the data for Roll no = "+RollNo);

String query = "DELETE FROM `rollstudent` WHERE RollNo = "+RollNo+";";

Statement st = null;

try {

st = con.createStatement();

} catch (SQLException throwables) {

throwables.printStackTrace();

System.out.println("Error in statement");

}

int affected = 0;

try {

affected = st.executeUpdate(query);

} catch (SQLException throwables) {

throwables.printStackTrace();

System.out.println("Error in deleting from table");

}

System.out.println( affected + " rows were affected");

labres.setText( affected+" rows were affected ");

tf1.setText("");

}

if (e.getSource()==update){

int RollNo = Integer.parseInt(tf1.getText());

String name = tf2.getText();

int age = Integer.parseInt(tf3.getText());

String gender = tf4.getText();

String address = tf5.getText();

int mark1 = Integer.parseInt(tf6.getText());

int mark2 = Integer.parseInt(tf7.getText());

int mark3 = Integer.parseInt(tf8.getText());

int mark4 = Integer.parseInt(tf9.getText());

int mark5 = Integer.parseInt(tf10.getText());

double CGPA = Student.CGPACalculate(mark1,mark2,mark3,mark4,mark5);

String query = "UPDATE `rollstudent` SET `name`=\""+name+ "\",`age`=\""+age+"\",`gender`=\""+gender+"\",`address`=\""+address+"\",`mark1`=\""+mark1+"\",`mark2`=\""+mark2+"\",`mark3`=\""+mark3+"\",`mark4`=\""+mark4+"\",`mark5`=\""+mark5+"\",`CGPA`=\""+CGPA+"\" WHERE RollNo = "+RollNo+";";

Statement st = null;

try {

st = con.createStatement();

} catch (SQLException throwables) {

throwables.printStackTrace();

System.out.println("Error in statement");

}

int affected = 0;

try {

affected = st.executeUpdate(query);

} catch (SQLException throwables) {

throwables.printStackTrace();

System.out.println("Error in entering into table");

}

System.out.println(affected + "rows were affected");

labres.setText(affected + "rows were affected ");

tf1.setText("");

tf2.setText("");

tf3.setText("");

tf4.setText("");

tf5.setText("");

tf6.setText("");

tf7.setText("");

tf8.setText("");

tf9.setText("");

tf10.setText("");

}

if (e.getSource()==end){

try {

st.close();

} catch (SQLException throwables) {

throwables.printStackTrace();

}

try {

con.close();

} catch (SQLException throwables) {

throwables.printStackTrace();

}

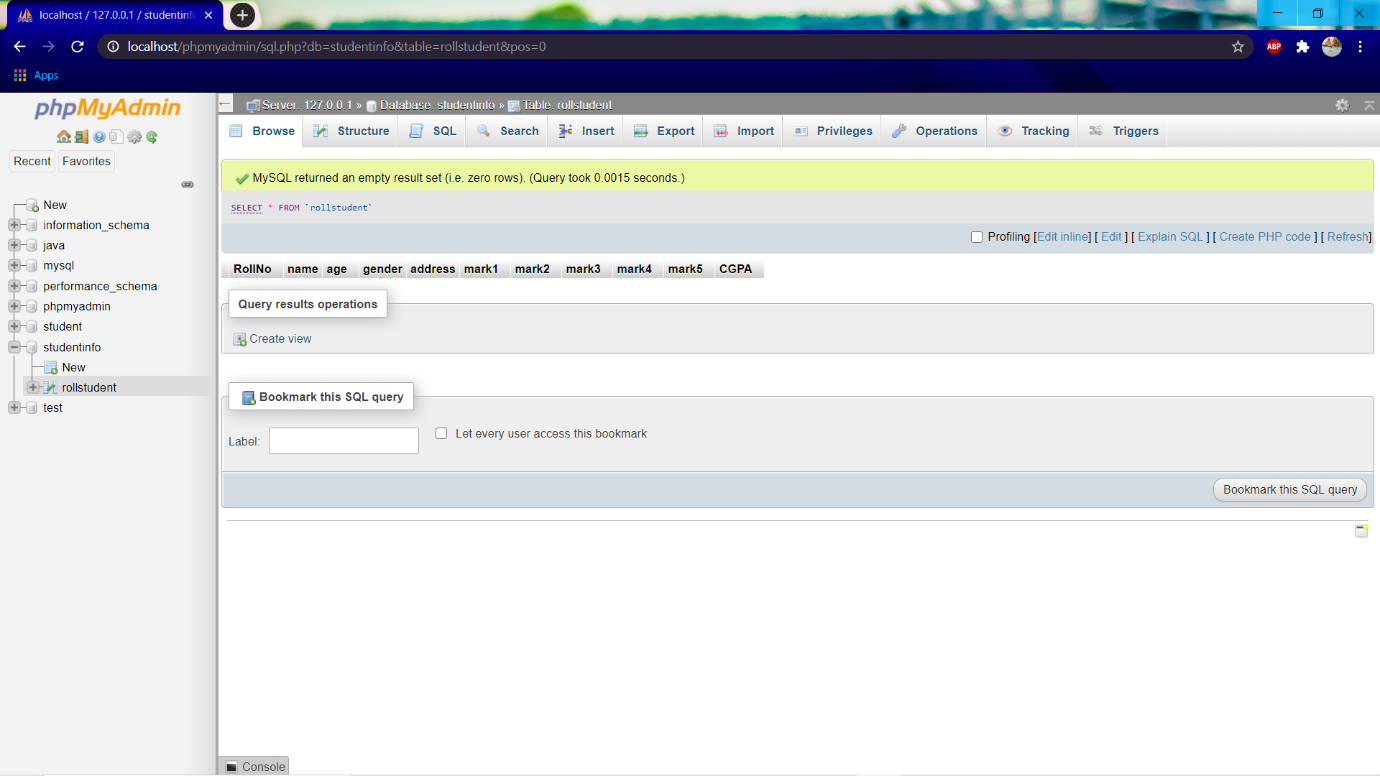
}

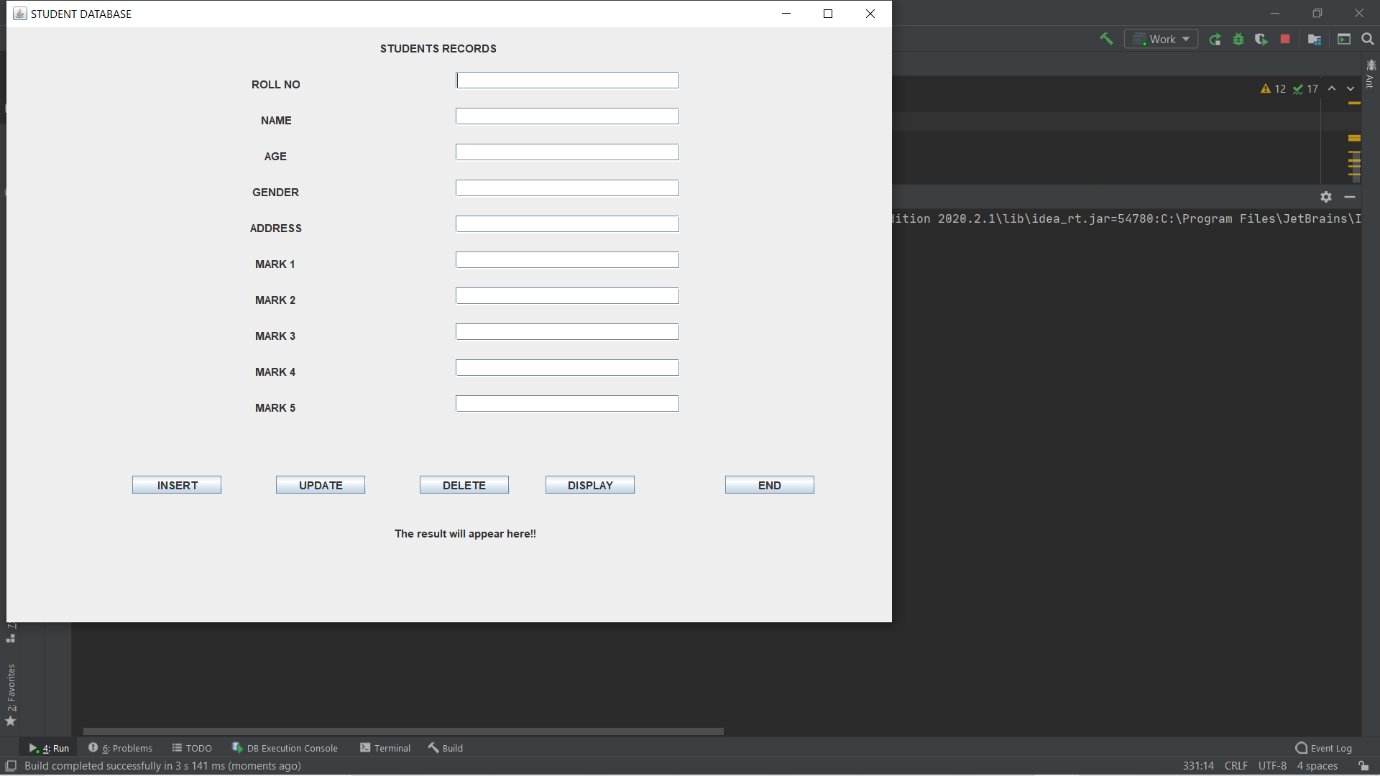
}

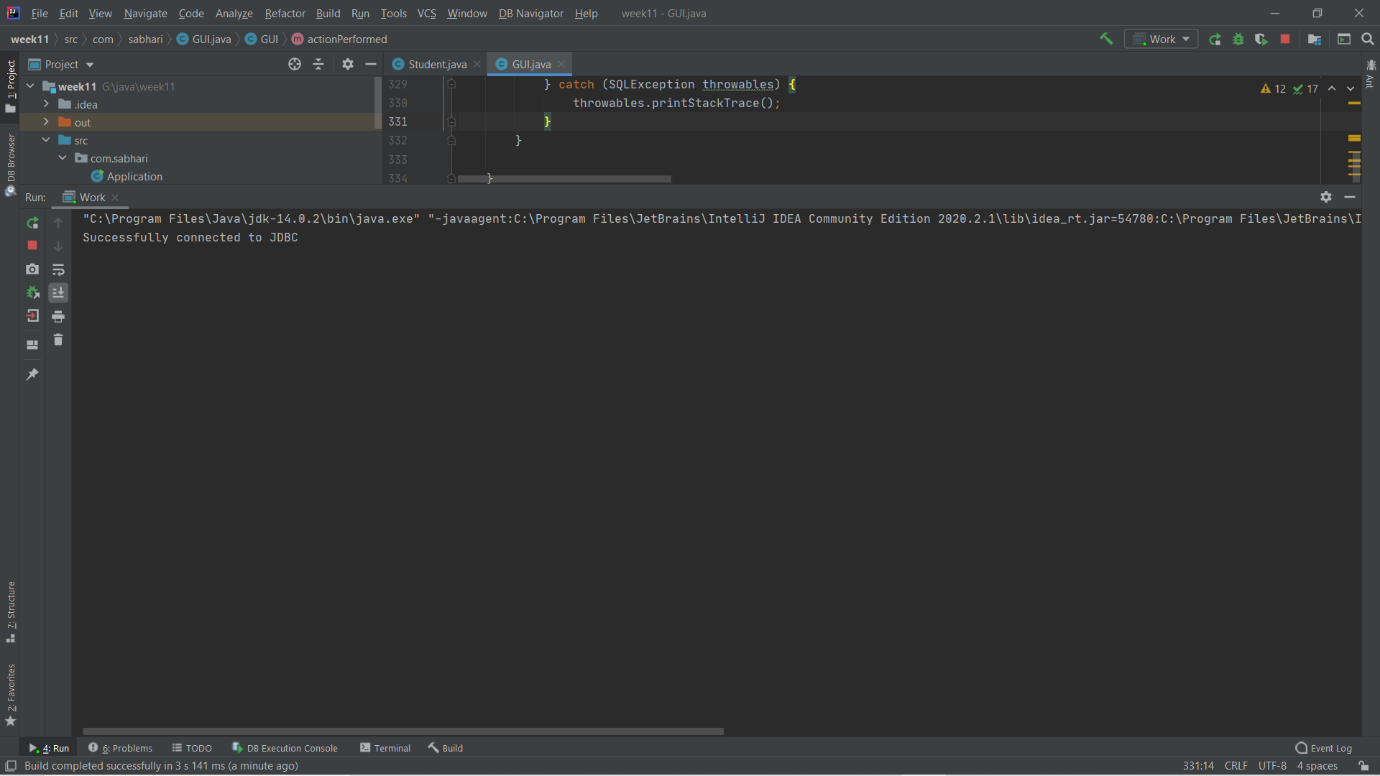
}

The above program combines the GUI with JDBC server and works on a relational database. In my program I had used XAMPP because its my favourite relational database. Lets jump in to examine the execution of the above code…

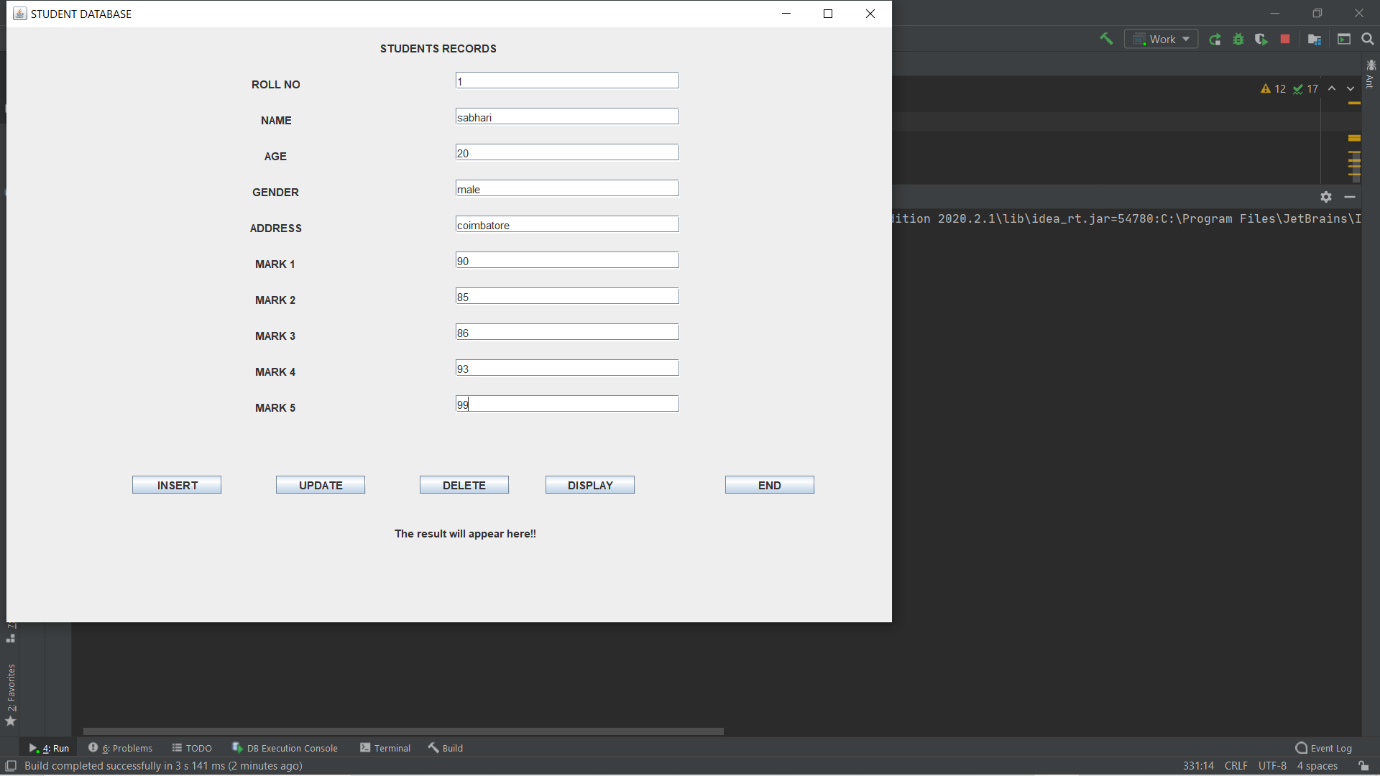
**OUTPUT:**

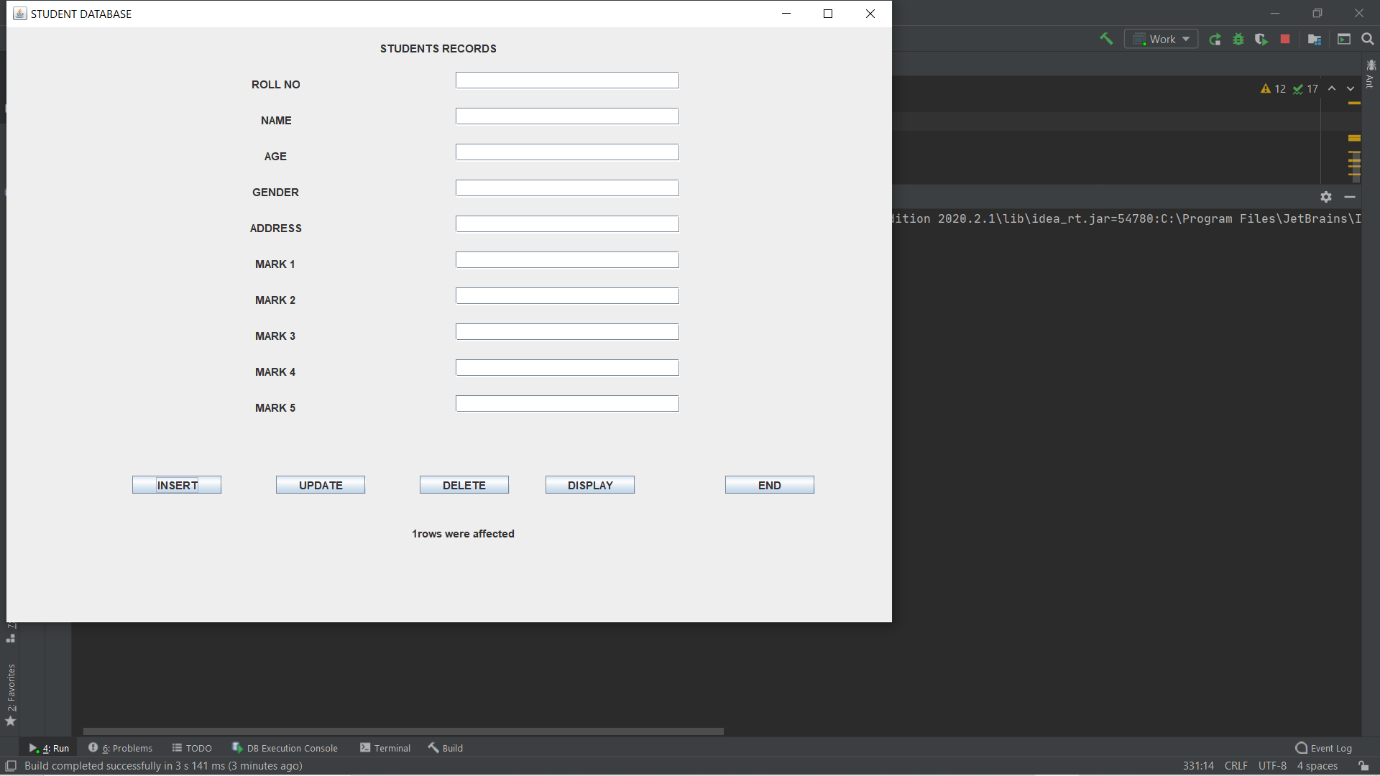
As a initial step we create a data base named studentinfo which contains the table named rollstudent which has the following represented columns. We made roll no as a primary key to optimally use the power of relational database. The picture after the initial setup is displayed below.

 Now let us compile and run the program. The GUI looks like below.

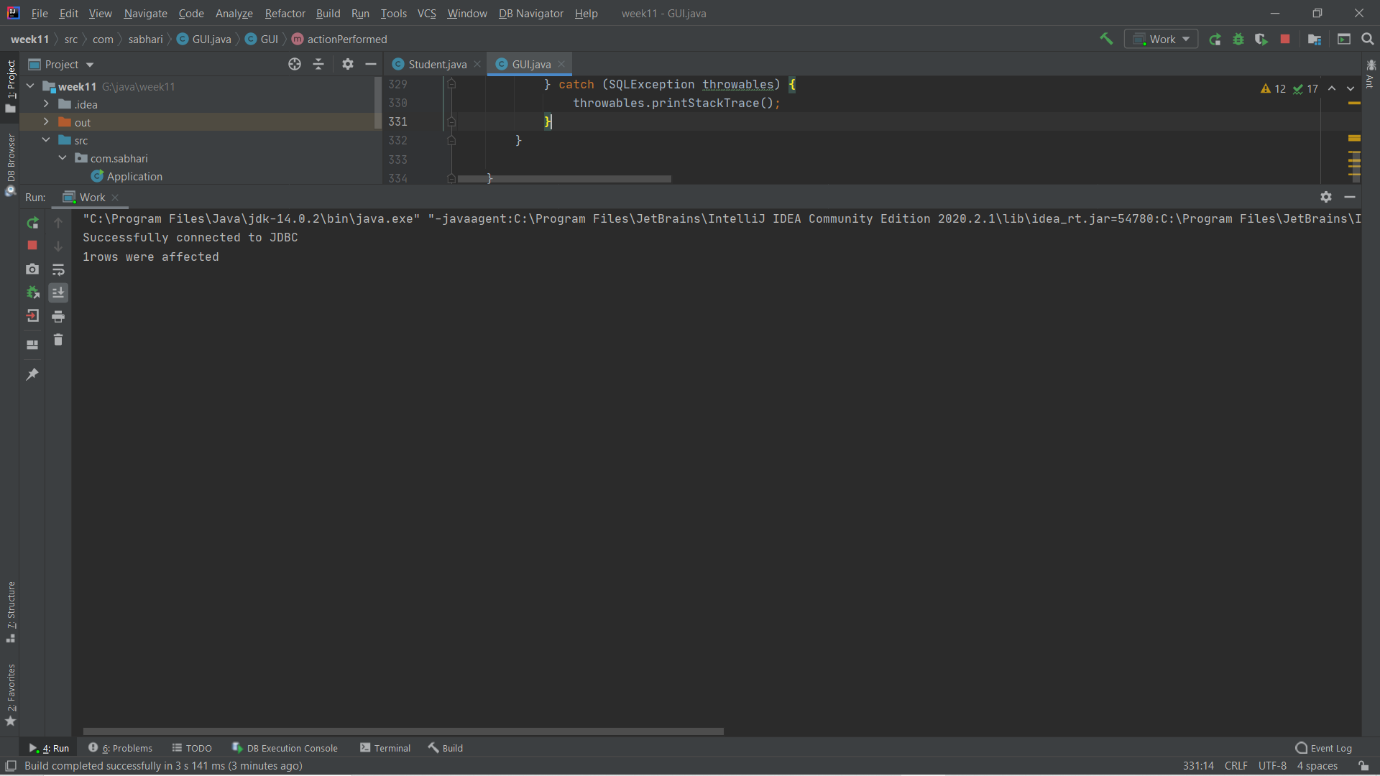
We also check the console.

As of now we don’t have any problem. Our application is running without any error. And also the JDBC connection is success.

We now enter our first record about a student to be placed in our database.

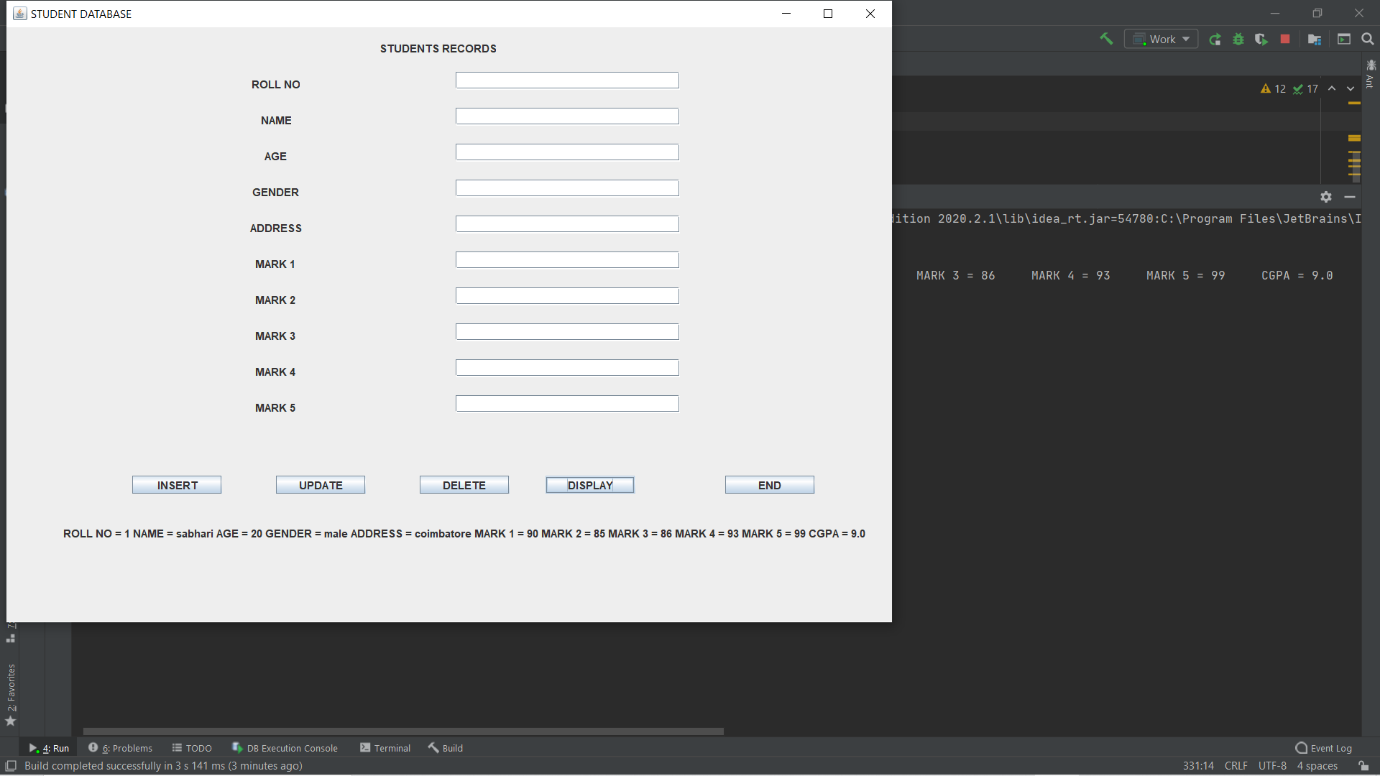
We press INSERT button here.

We got a message on 1 row were affected which says the success of program in entering data to a table.

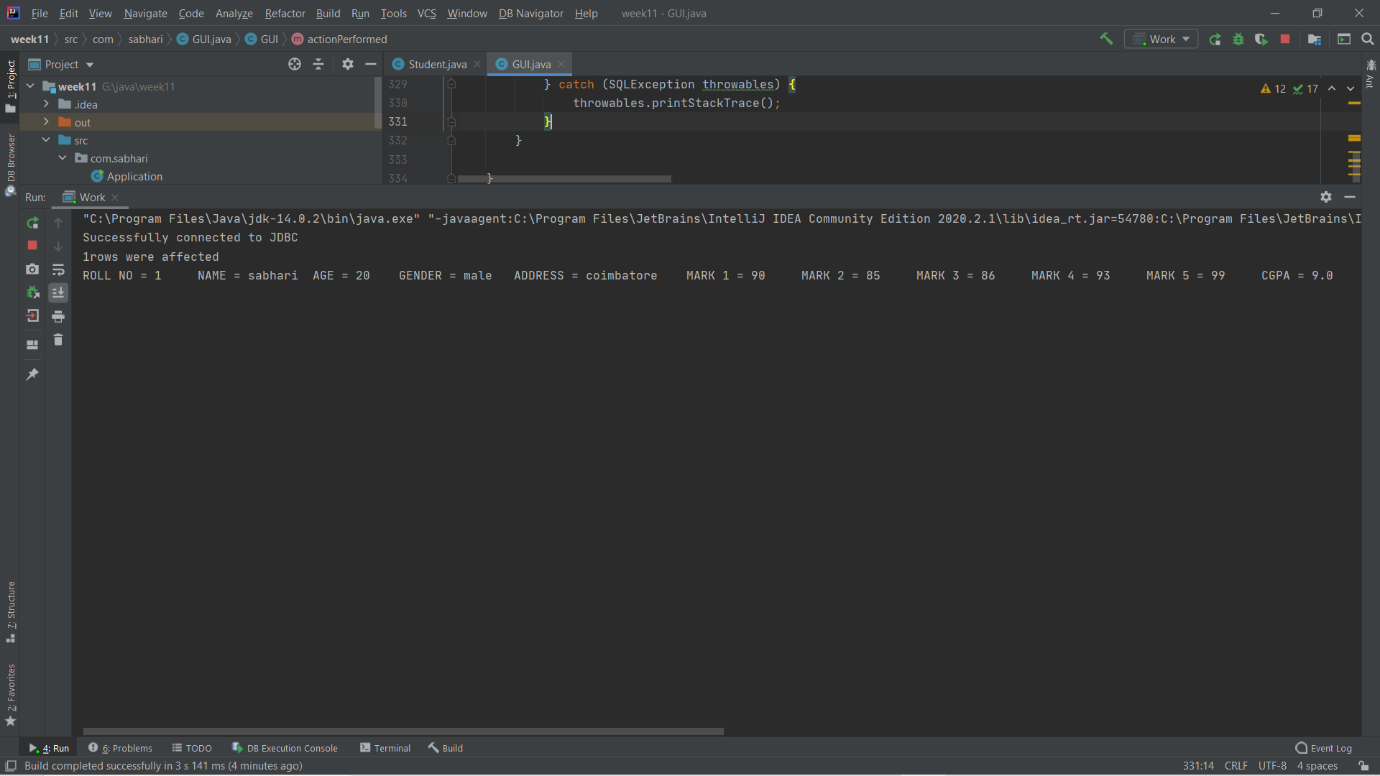
We crosscheck it with our console window.

Thus the first record is inserted without any error.

Let us try to display the record by clicking DISPLAY button.

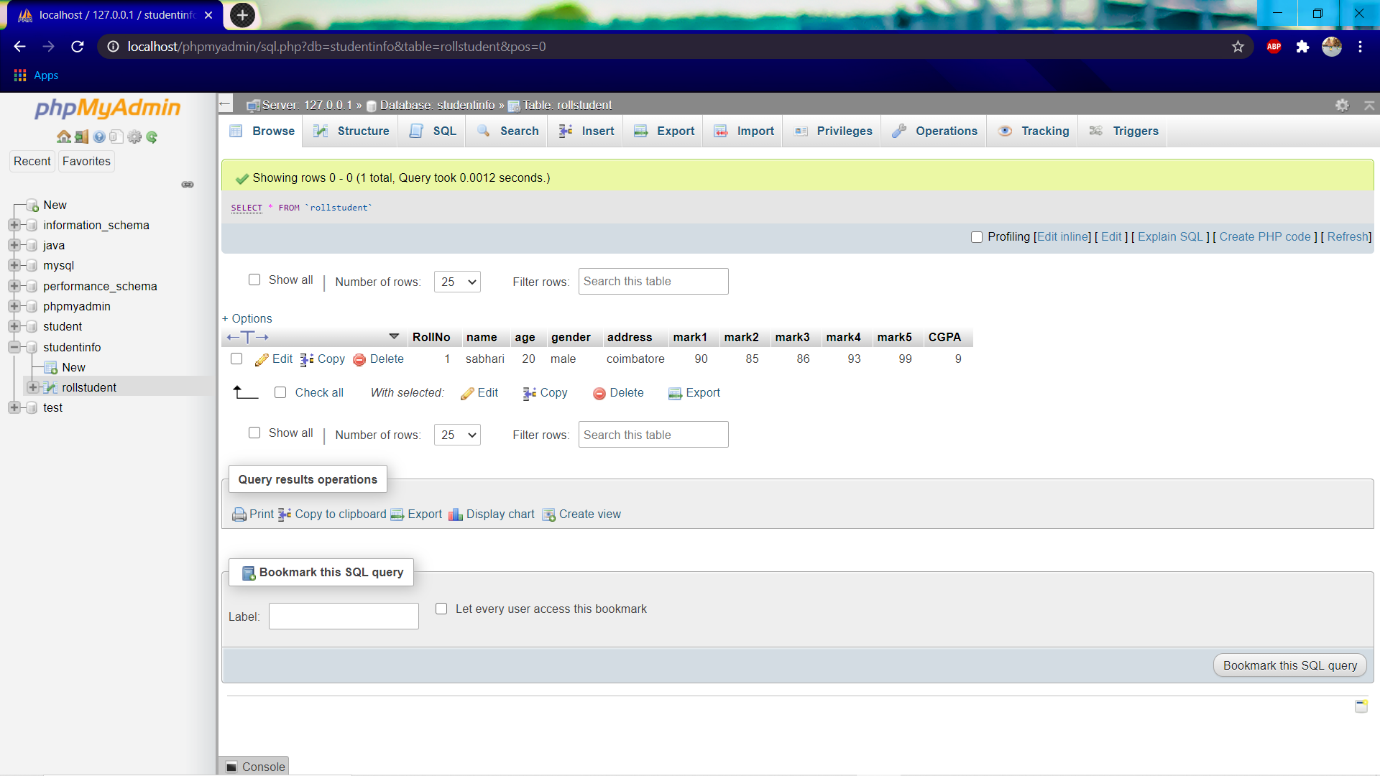


As expected the result is displayed.

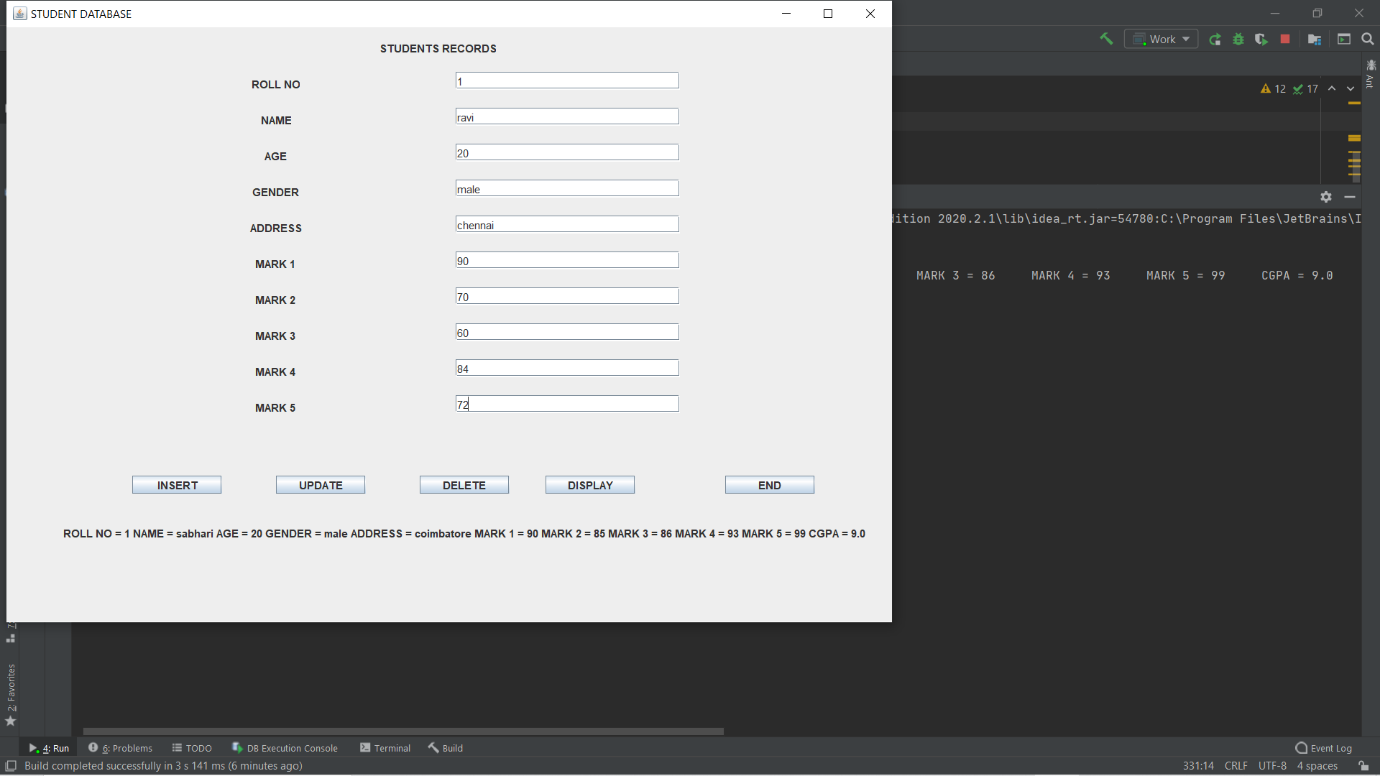
Let us cross check it with our console.

Works so fine and so good. That the table entry is displayed.

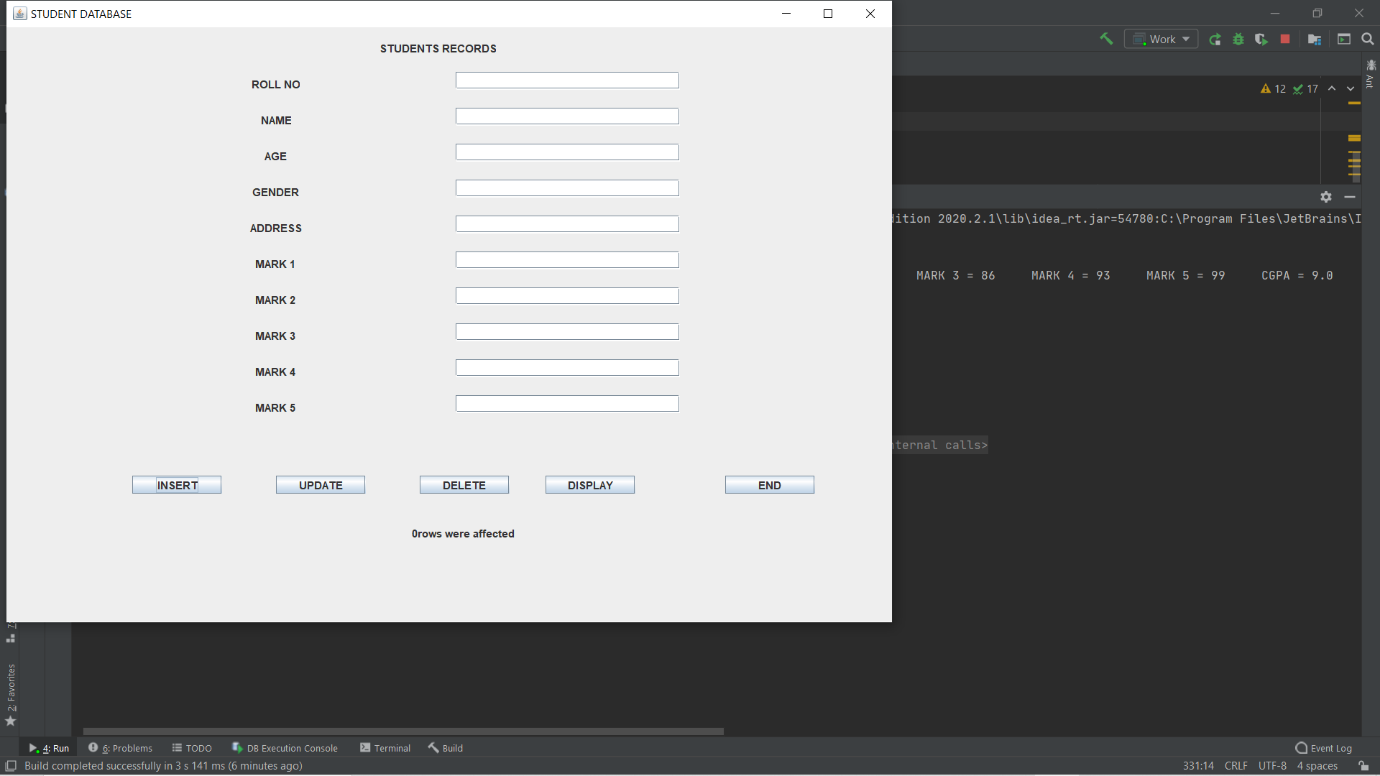
You may get doubts like, whether the record is inserted into the table? Whether it works correct? Whether it calculates the CGPA? The answer is below



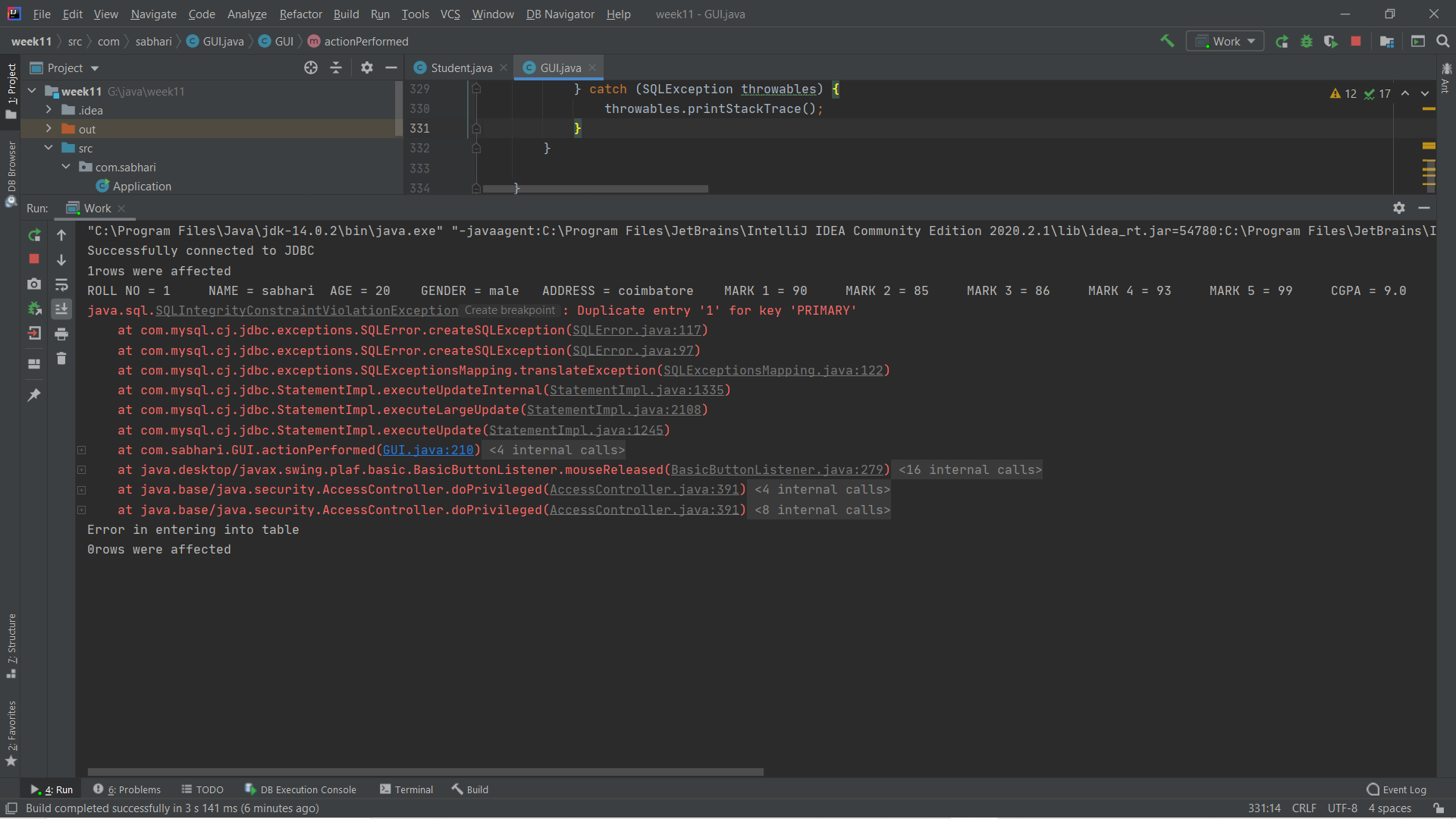
It works so good . the record is inserted into database and also viewed from database. So far it works so good.

Now let us try to insert a duplicate value for rollno 1 which is already existing.

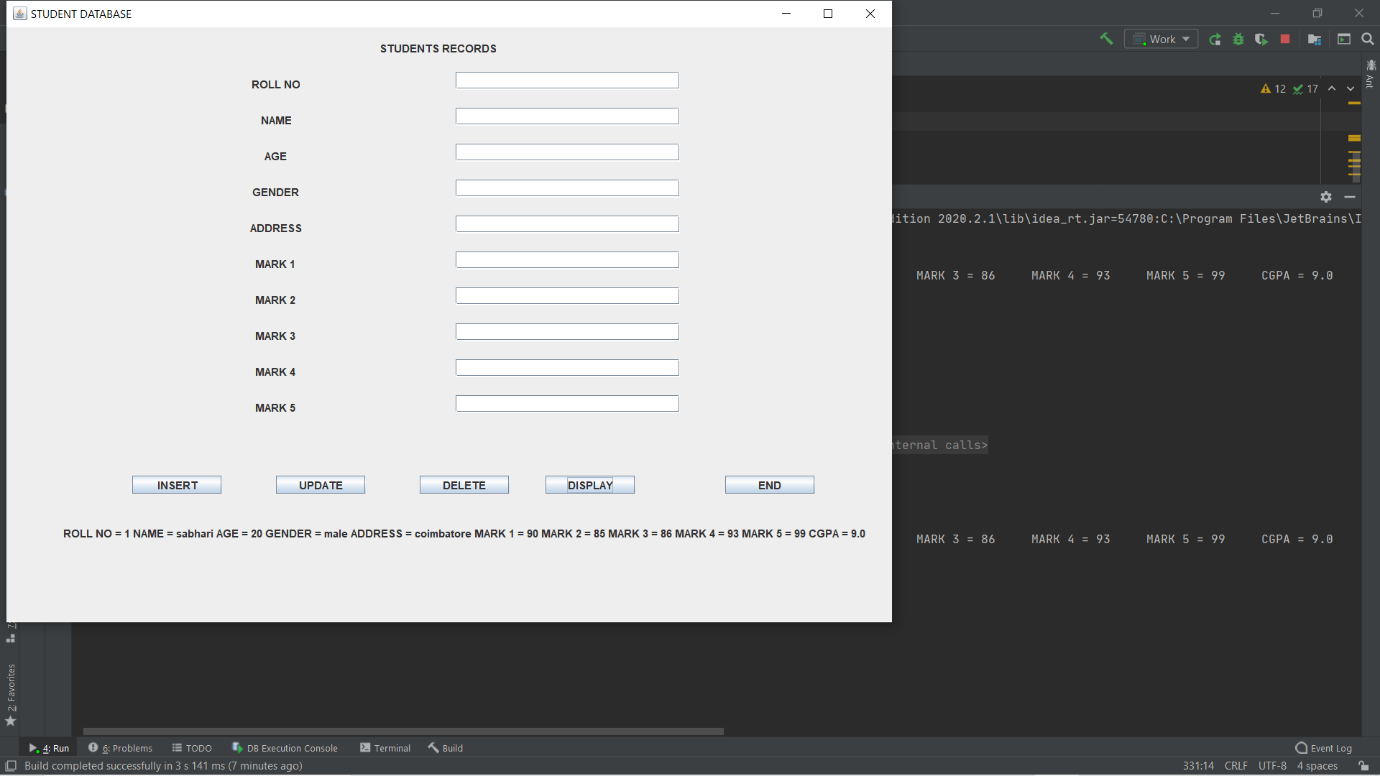
We have entered the duplicate values as shown above.

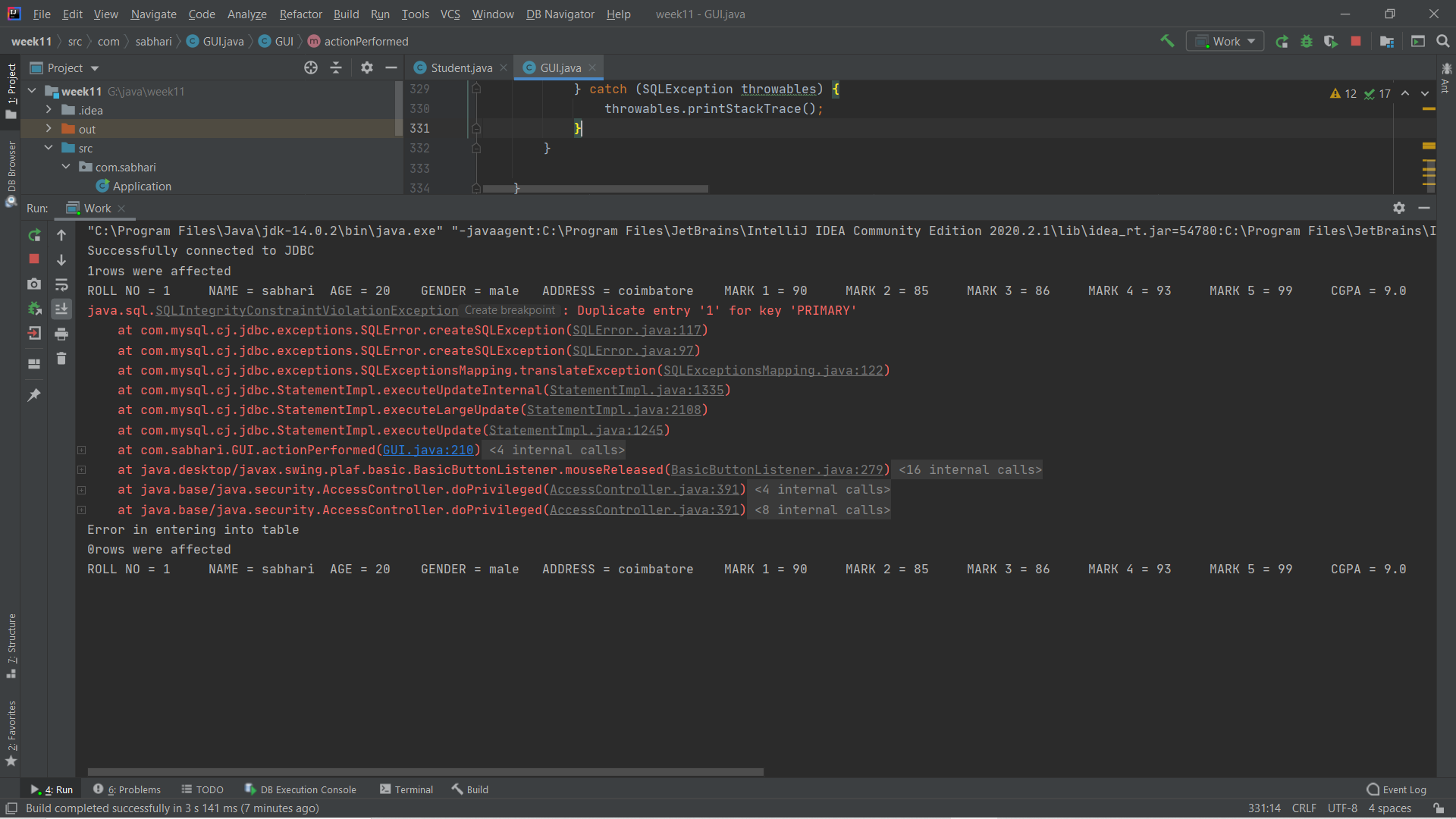
Let’s press insert.

Its says 0 rows were affected. It means that there is no change made to database. Because there may be an error. Lets find it out.

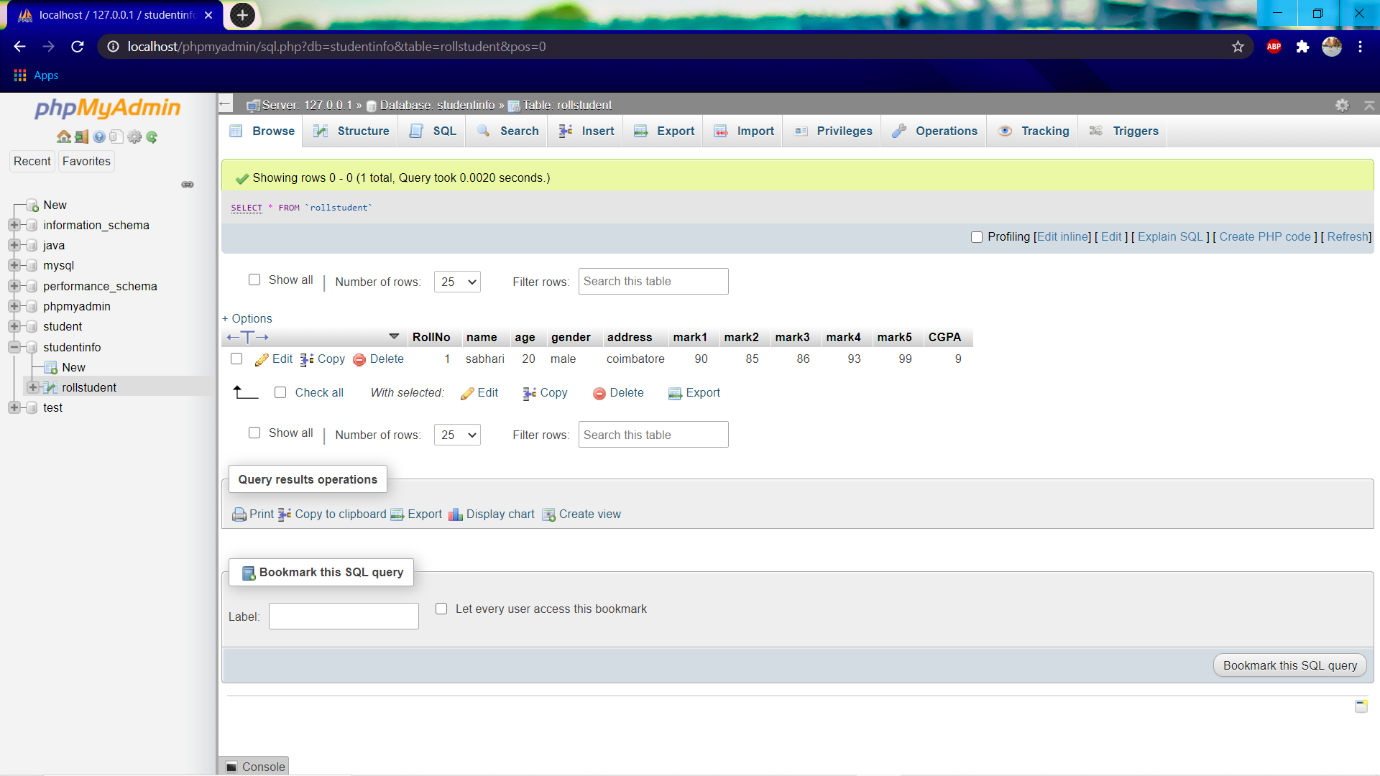
Now we look to our console.

It had replied with a safe exception rather than hard stopping of program. The exception says us Duplicate entry 1 for primary key. Thus we came to know that we cannot add duplicate entries to database and make it inconsistent. It works so good so far. Lets explore more.

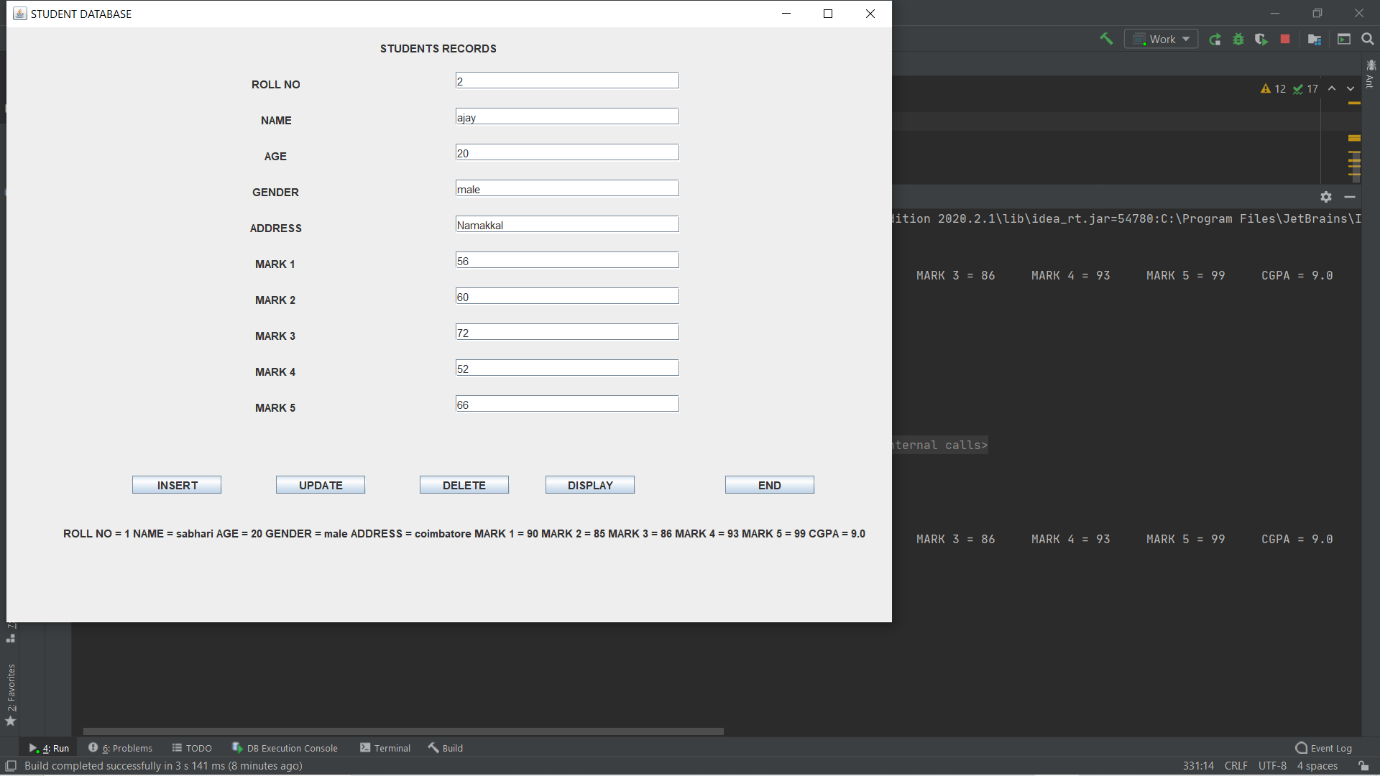
Lets view whether the duplicate entry had affected our database.

Lets check with console too.

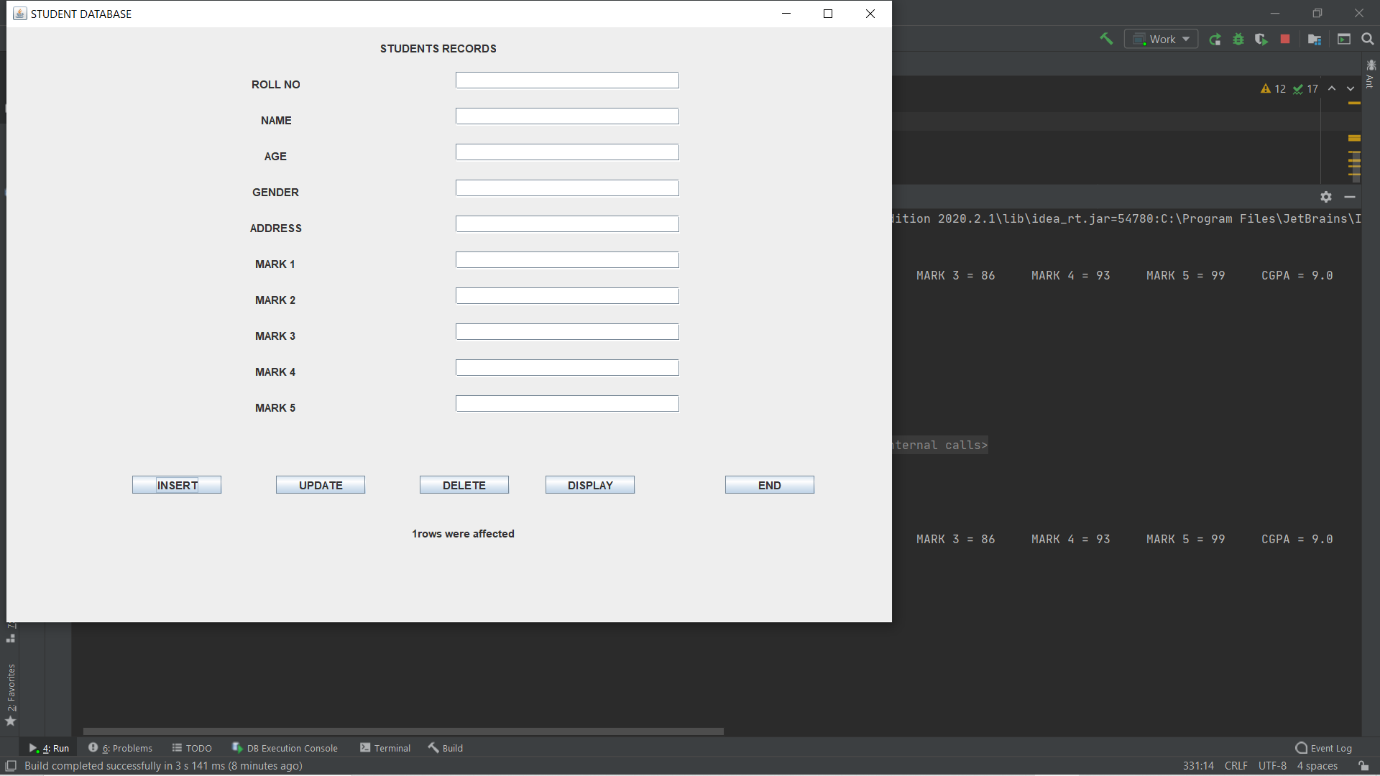
Thus we see that the duplicate value has not affected the database.

Let us check the database directly.

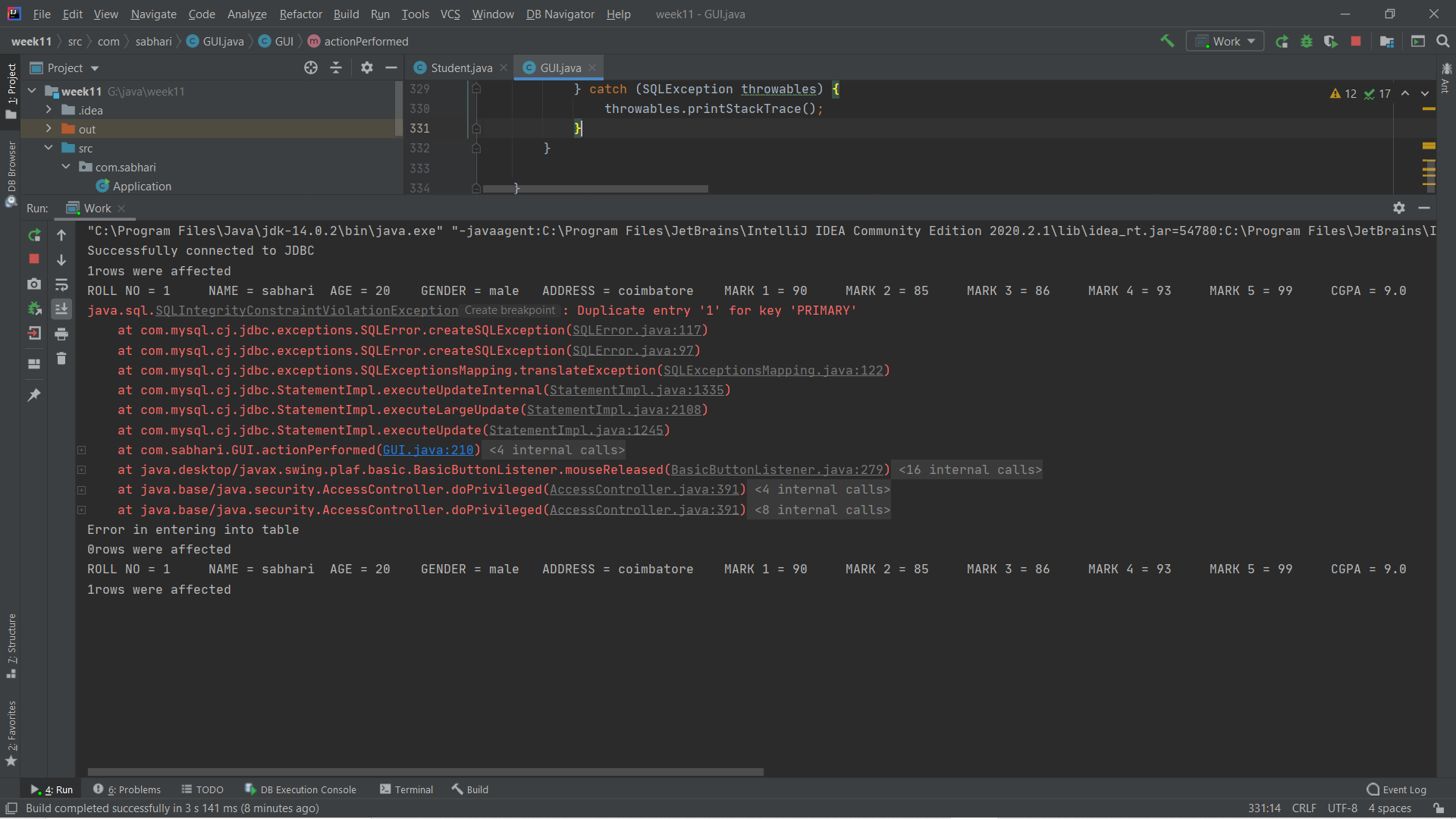
Our database is not affected.

Let us add another data to database.

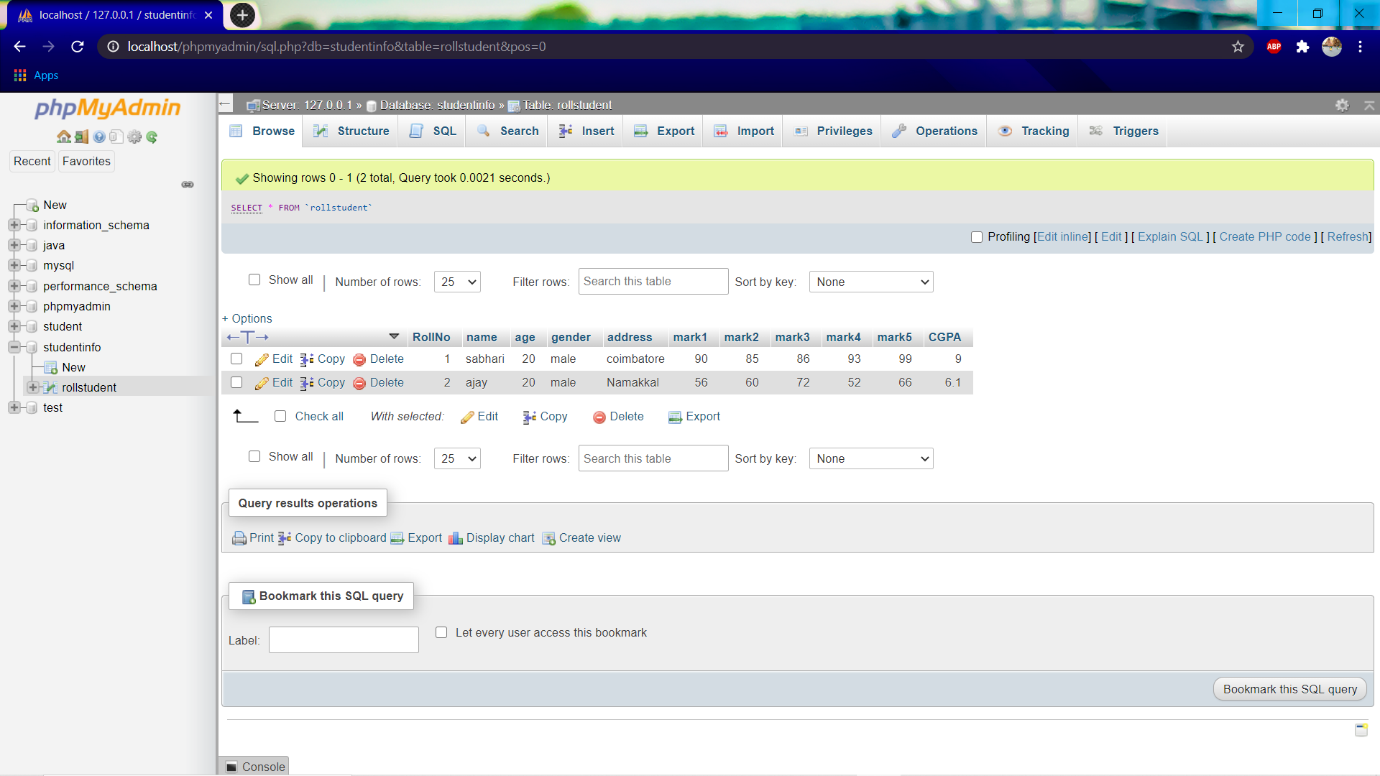
Lets press INSERT.



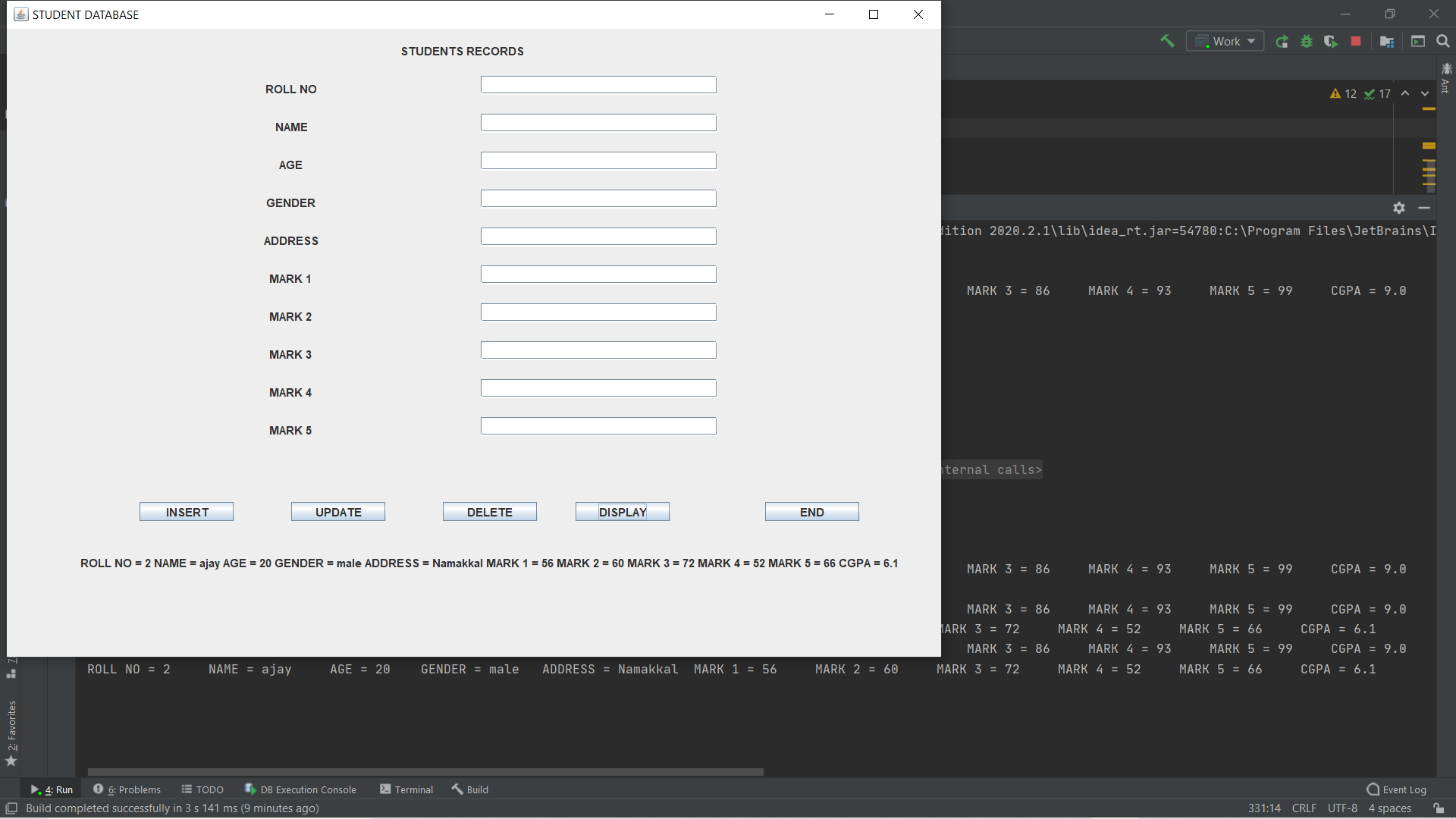
1 row affected message indicates success.

Lets us check it with console.

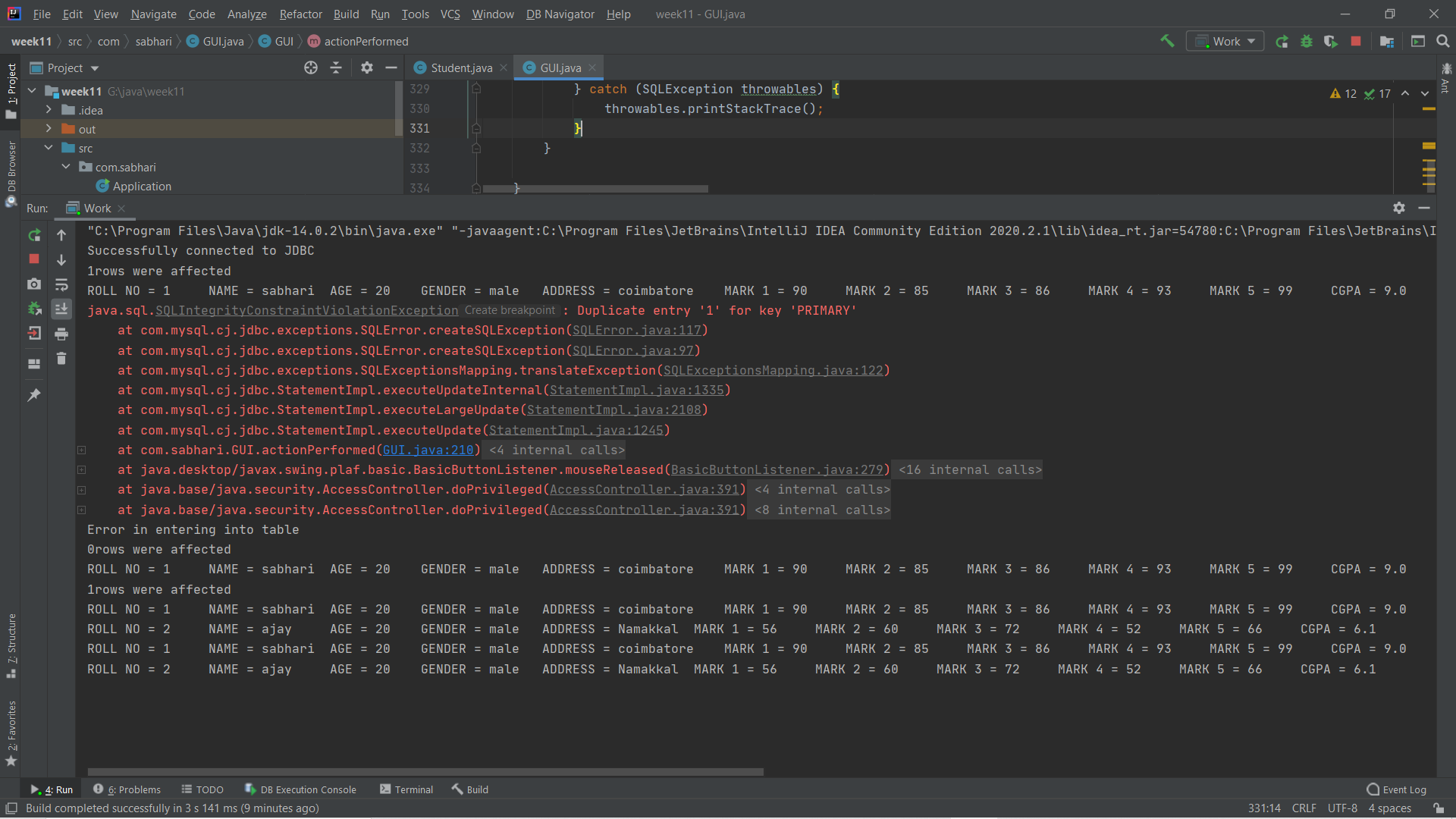
It had proceeded without any error.

Let us check the data base directly.

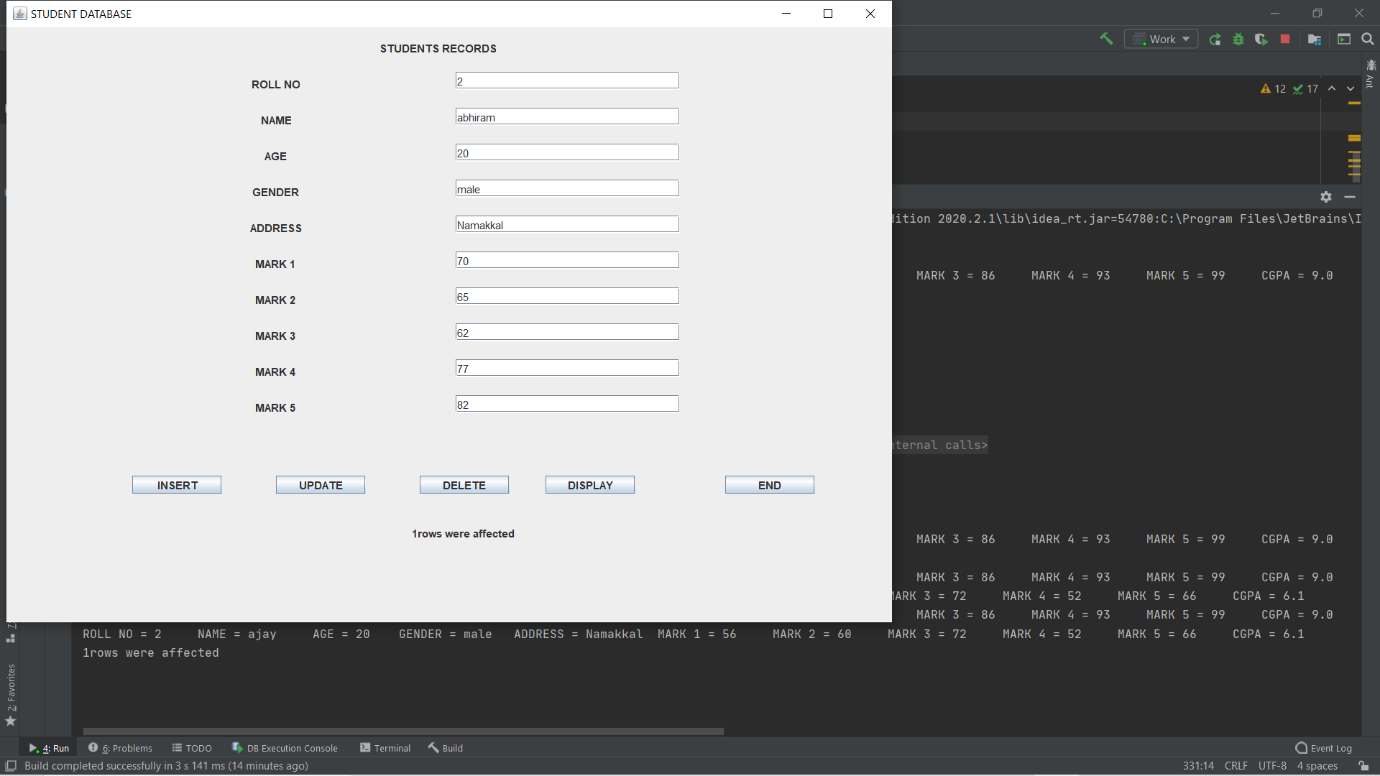
It works cool, it had also computed the CGPA based on the marks.

Now we view it using DISPLAY

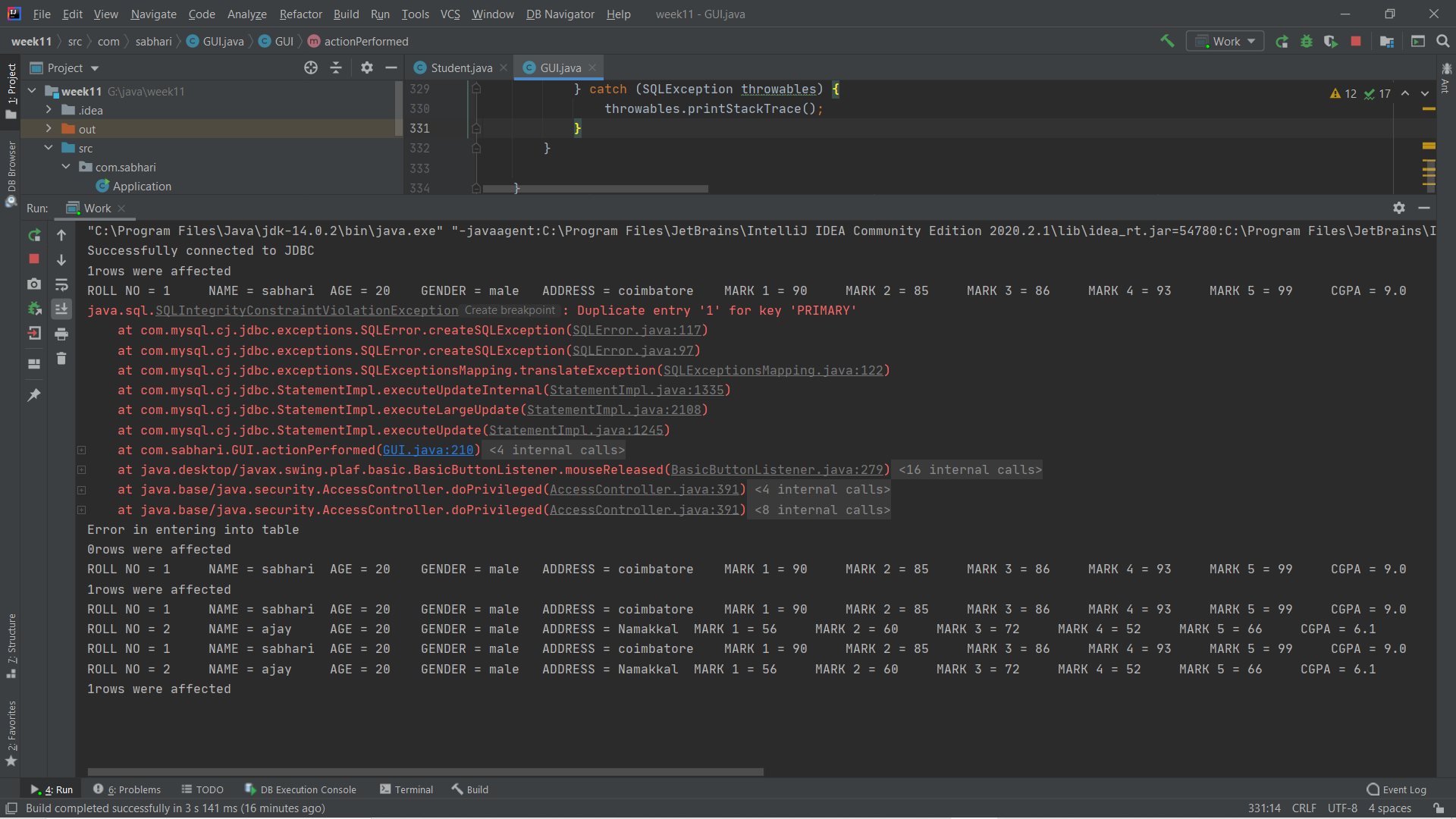
It had displayed the last over written record. Lets hit console for all records.



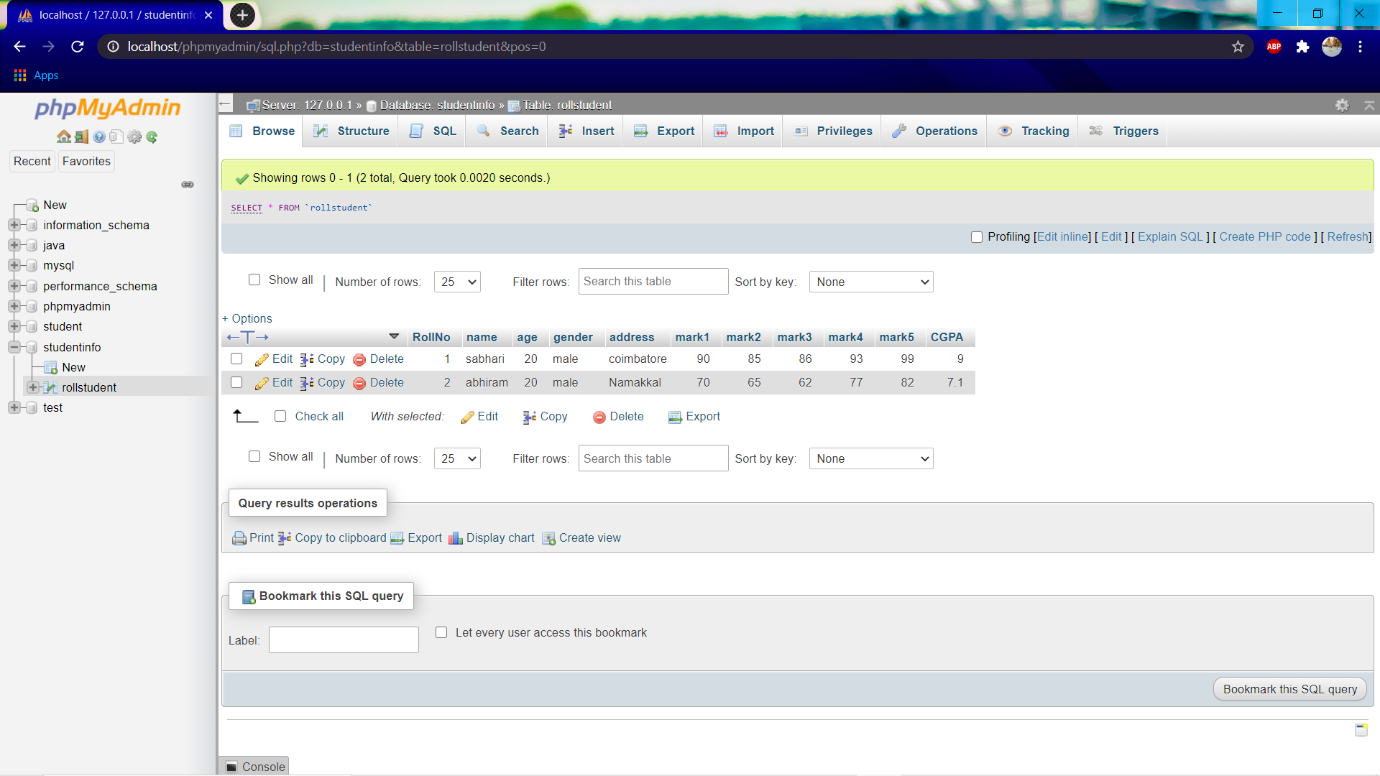
Now the database has two records and they are displayed. I had accidently pressed the display twice , so it gets executed twice.

Now let us try to update the existing values. The roll no 2 is being updated.we enter the updation values as below.

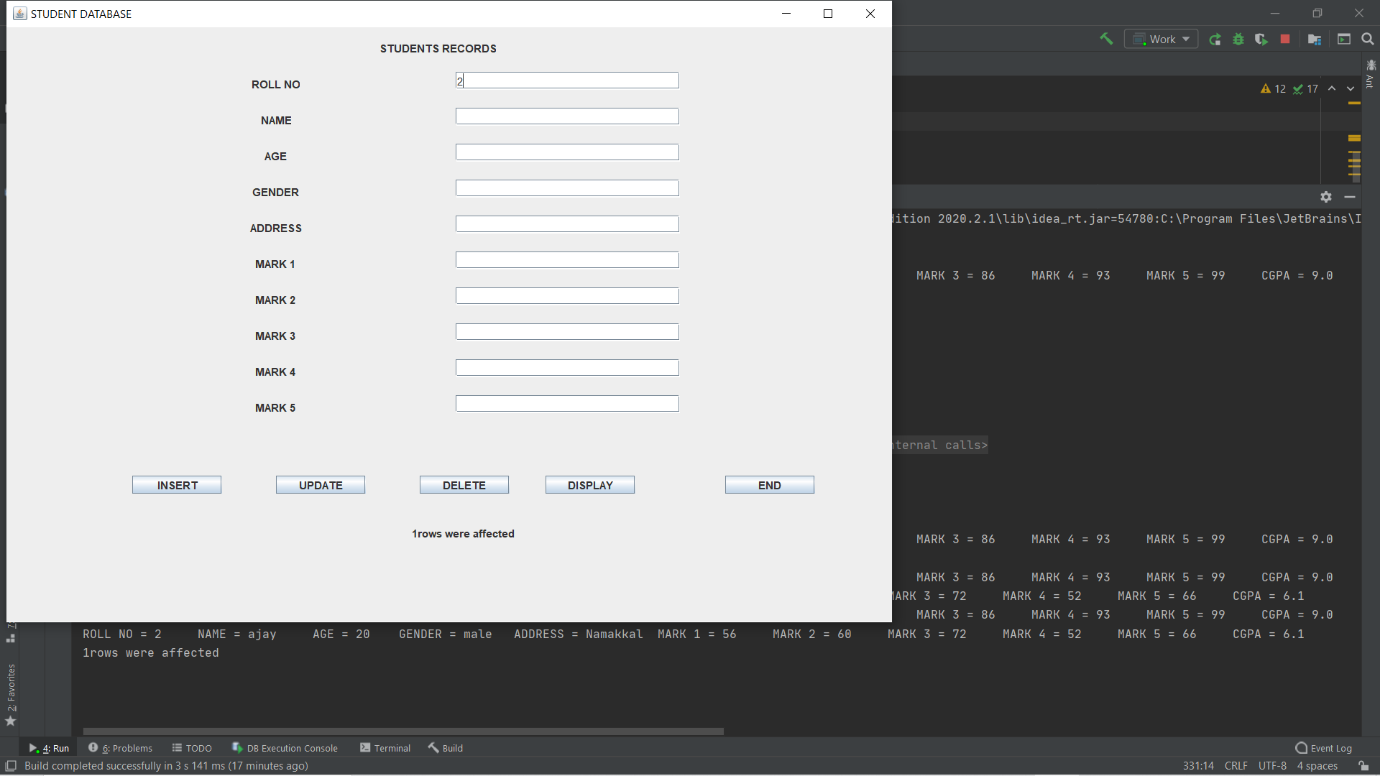
As we press update we will get a 1 roe affected message.

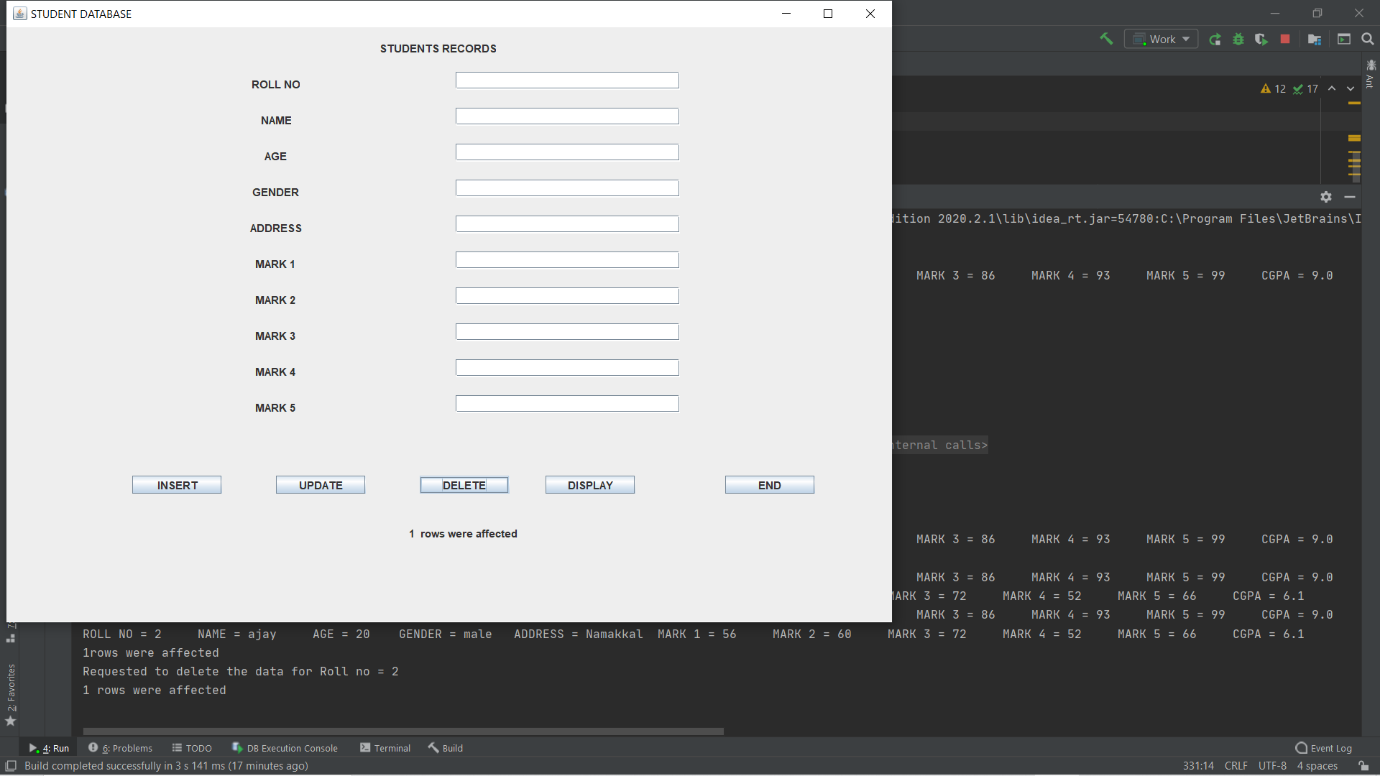
We will directly head to console.

The message says that one row had been successfully updated.

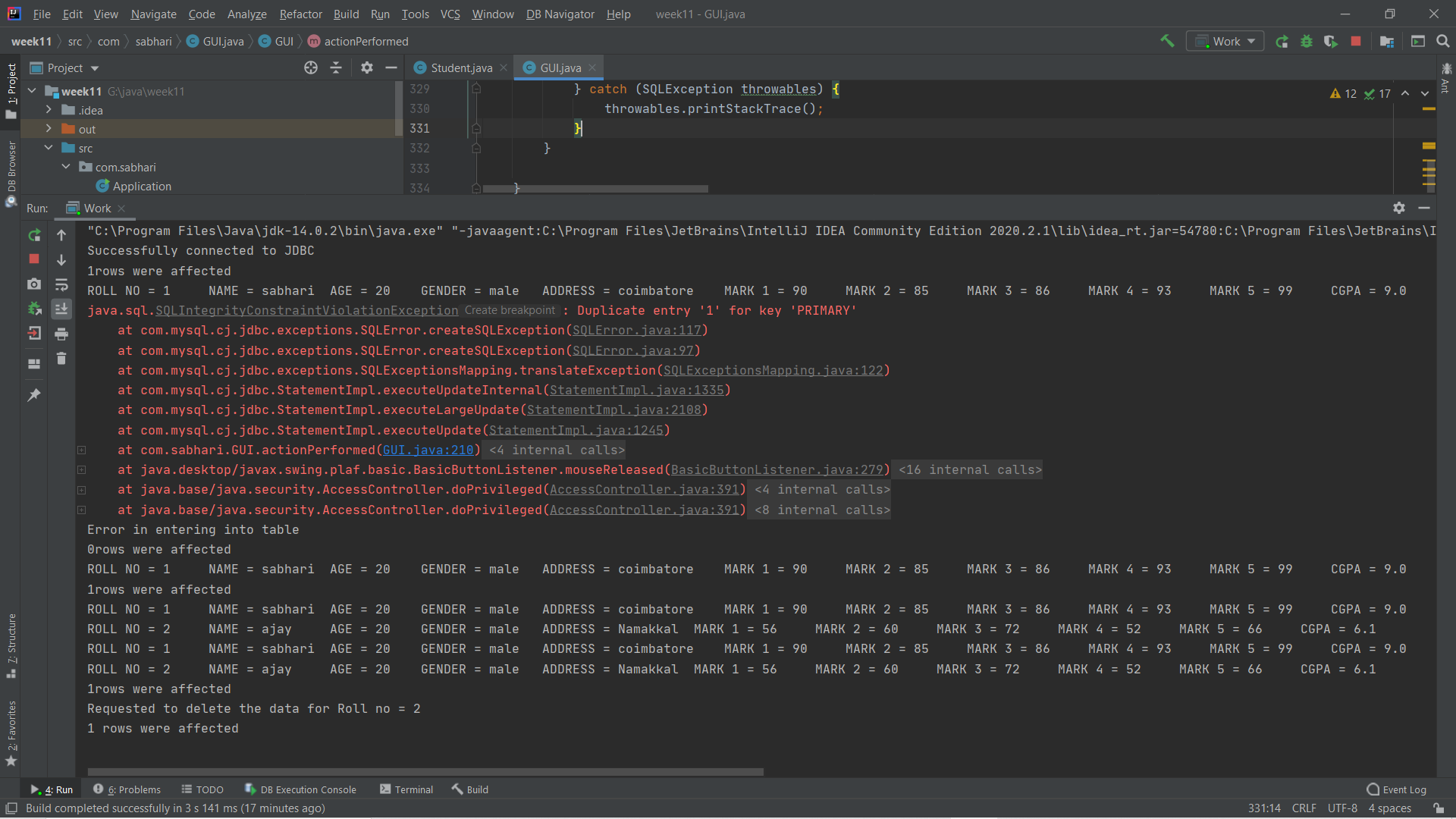
Let us check it in our database.

The value has been updated in our database.

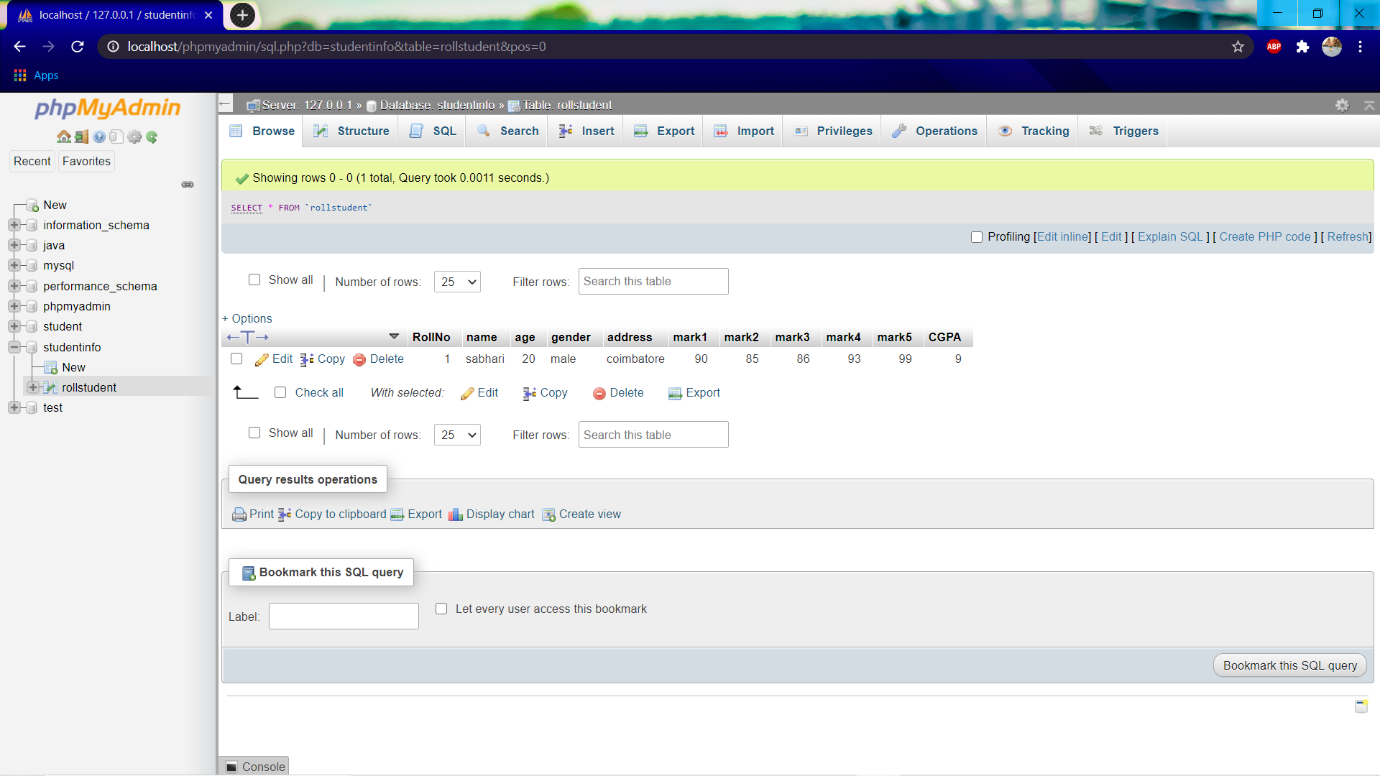
Now lets try to delete the record. We had entered 2 in roll number field.

Now we press DELETE button.

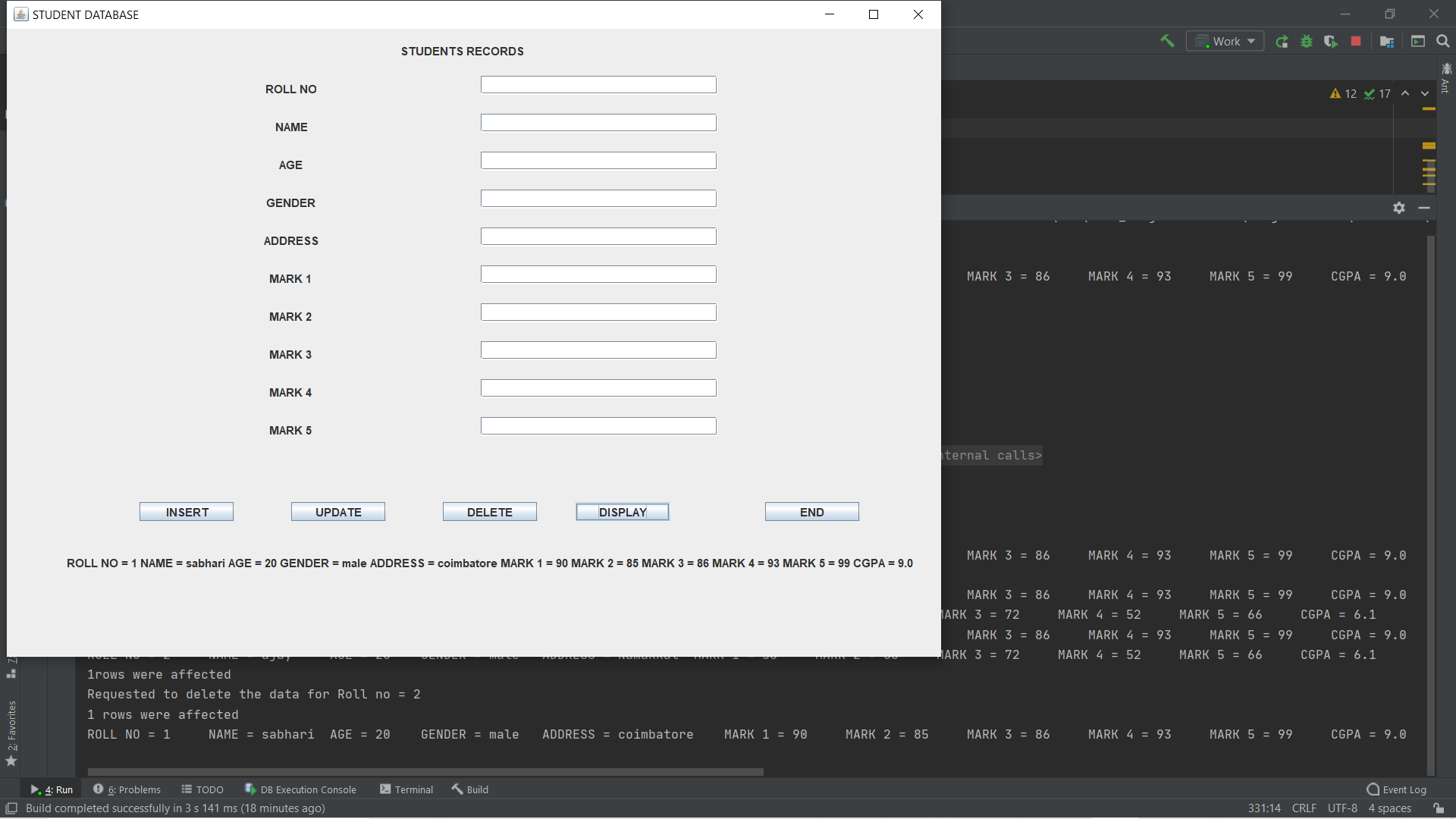
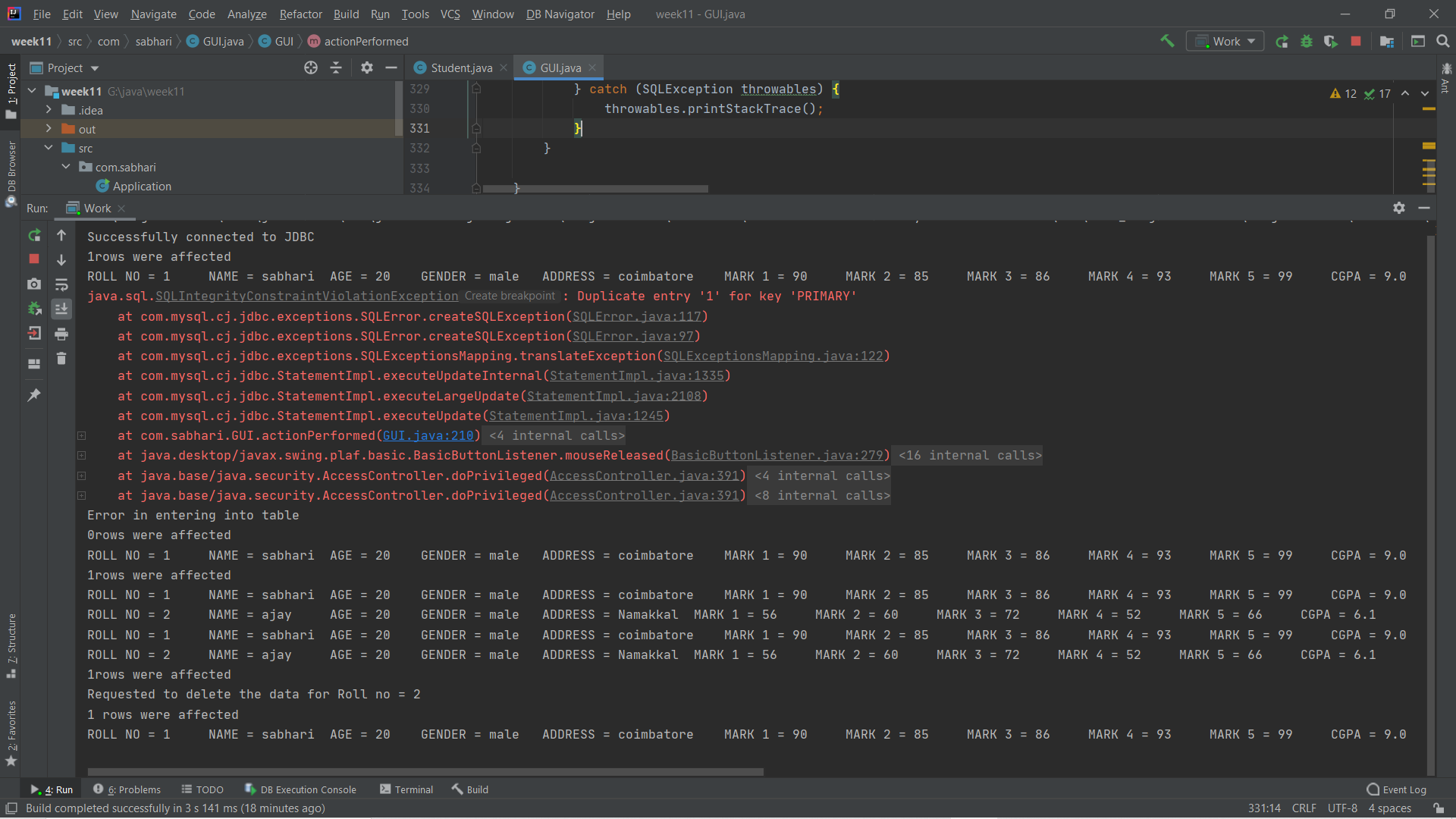
The message says that the row has been deleted successfully.

Lets us check it with our console.

The console says that we have requested to delete the record of roll no 2 and it had been deleted successfully.

Lets look at database for confirmation.

It got deleted. Thus our code is working completely correct. We will use display button.



The display function also return the correct result.

**Thus our code is working completely correct.**