2.1. Do the workshop 2 in the CD (JDBC-II).

2.2. Write a Java application, **`StudentSearch`** that searches for a given record in the database table, **`student`**. The record details should be displayed in the command prompt.

Column Name	Data Type
Name	Varchar
Rollno	Numeric
Class	Varchar

Solution (type the below code):

```
import java.awt.*;
import java.awt.event.*;
import java.sql.*;
import java.util.*;
import javax.swing.*;

class StudentSearch extends JFrame implements ActionListener
{
    JLabel lblName = new JLabel("Name");
    JLabel lblRollNo = new JLabel("RollNo");
    JTextField txtName = new JTextField(20);
    JTextField txtRollNo = new JTextField(20);

    JButton btnSearch = new JButton("Search");
    JButton btnExit = new JButton("Exit");

    Statement  st;
    String strSql;
    JPanel pnlStart,pnlEnd;

    public StudentSearch(String title)
    {
```

```
super(title);
try
{
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
}
catch(ClassNotFoundException ce)
{
    System.out.println(ce);
}
try
{
    String url = "jdbc:odbc:addr";
    Connection con = DriverManager.getConnection(url);
    st = con.createStatement();
}
catch(SQLException ce)
{
    System.out.println(ce);
}

getContentPane().setLayout(new BorderLayout());

pnlStart = new JPanel();
pnlStart.setLayout(new GridLayout(3,2));

pnlStart.add(lblName);
pnlStart.add(txtName);

pnlStart.add(lblRollNo );
pnlStart.add(txtRollNo);
```

```
        pnlEnd = new JPanel();

        pnlEnd.add(btnSearch);
        pnlEnd.add(btnExit);

        btnSearch.addActionListener(this);
        btnExit.addActionListener(this);

        getContentPane().add(pnlStart,BorderLayout.PAGE_START);
        getContentPane().add(pnlEnd,BorderLayout.PAGE_END);

        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        setSize(300,200);
        setVisible(true);
    }

    public void actionPerformed(ActionEvent e)
    {
        try
        {
            if(e.getSource().equals(btnSearch))
            {
                int rollno = Integer.parseInt(txtRollNo.getText());
                String strSql = "select * from student where name like'" +
txtName.getText();
                strSql = strSql + "' and rollno=" + rollno;
                ResultSet rs = st.executeQuery(strSql);
                if(rs.next() == false)
                    System.out.println("Record not found");
                else
                {
```

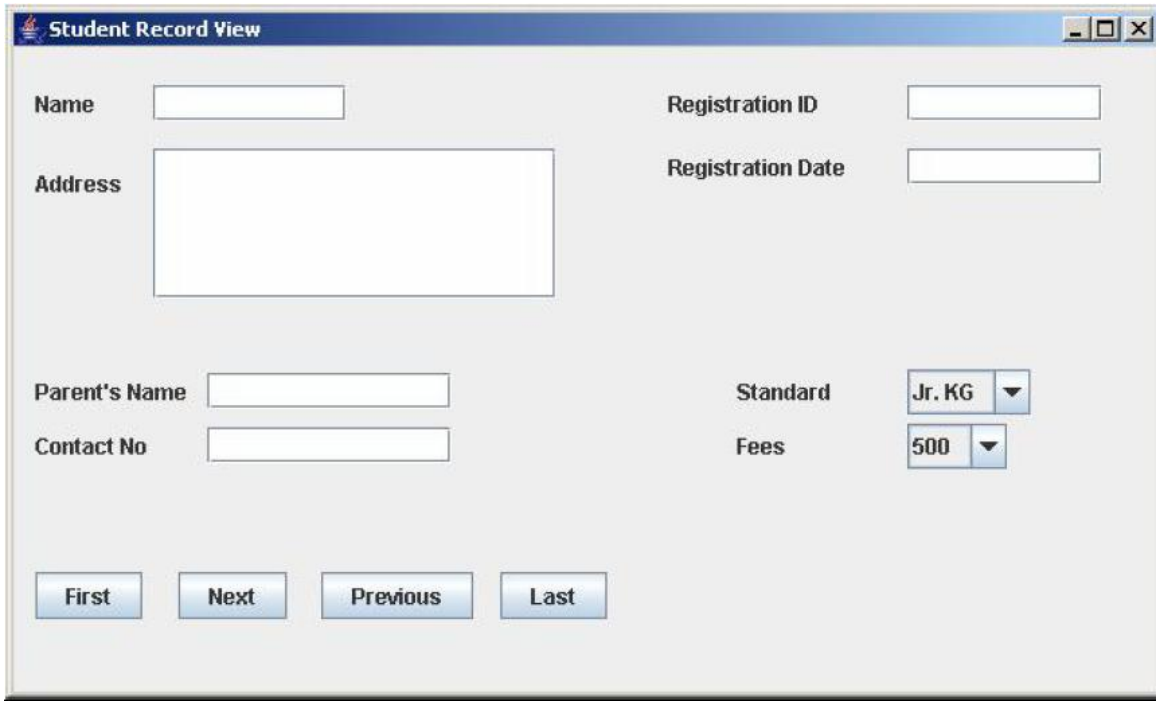
```
                    System.out.print("\nName : ");
                    System.out.print(rs.getString(1) + "\t");
                    System.out.print("\nRollNo : ");
                    System.out.print(rs.getInt(2) + "\t");
                    System.out.print("\nClass : ");
                    System.out.print(rs.getString(3));
                }
                rs.close();
            }
            else if(e.getSource().equals(btnExit))
                System.exit(0);
        }
        catch(SQLException ce)
        {
            System.out.println(ce);
        }
    }

    public static void main(String args[ ])
    {
        StudentSearch objStudentSearch = new StudentSearch("Student");
    }
}
```

The output of the program

Do It Yourself

2.3. Write a Java application StudentDetails that will traverse records from a student table by using DisconnectedRowSet.

The output of the program is as shown as below.



2.4. Rewrite exercise 1.1 by using ConnectedRowSet

2.5. Apply for a policy that denied all programs accessing data from a student table is as shown as exercise 2.1.

2.6. Write an applet application that writes some text into a file. Write a policy file allowed to do that.

The output of the program is as shown as below.