STUDENT REGISTRATION FORM

```
package connection;

public interface DBInfo {
   String DB_DRIVER="com.mysql.jdbc.Driver";
   String DB_URL="jdbc:mysql://localhost:3306/student_info";
   String DB_USER="root";
   String DB_PWD="payal";
}
```

```
package connection;

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;

public class ConnectionProvider {

   public static Connection provideConnection()
   {
      Connection con=null;
      try
      {
         con= DriverManager.getConnection(DBInfo.DB_URL,DBInfo.DB_USER,DBInfo.DB_PWD);
         return con;
      }
      catch(SQLException e)
      {
         System.out.println(e);
      }
      return con;
   }
}
```

```
package guiForms;

import listeners.StudentListener;

import javax.swing.*;
import java.awt.*;

public class StudentForm {
   JPanel p;
   JLabel l;
   JButton add,update,remove,view,login,register;

   StudentListener stListener;

   public boolean isLoggedIn=false;
```

```
StudentForm()
{
   JFrame frame=new JFrame("Student Form");
   frame.setLayout(null);
   frame.setResizable(false);
   frame.getContentPane().setBackground(new Color(2,2,20));
   frame.setLocation(100,100);

   l=new JLabel("Select an option ");
   l.setForeground(Color.white);
   l.setBounds(150,50,200,50);
   l.setFont(new Font(Font.SANS_SERIF,Font.PLAIN,20));
   frame.add(l);

   p=new JPanel();
   p.setBackground(Color.gray);
   GridBagLayout g=new GridBagLayout();
   GridBagConstraints gc=new GridBagConstraints();
   p.setLayout(g);
   p.setBounds(150,100,200,200);

   stListener=new StudentListener();

   add=new JButton("Add Student");
   add.setBackground(Color.orange);
   add.addActionListener(stListener);

   update=new JButton("Update Details");
   update.setBackground(Color.orange);
   update.addActionListener(stListener);

   remove=new JButton("Remove Student");
   remove.setBackground(Color.orange);
   remove.addActionListener(stListener);

   view=new JButton("View Details");
   view.setBackground(Color.orange);
   view.addActionListener(stListener);

   Insets i=new Insets(5,5,5,5);
   gc.gridx=30;
   gc.insets=i;

   gc.gridy=30; p.add(add,gc);
   gc.gridy=40; p.add(update,gc);
   gc.gridy=50; p.add(remove,gc);
   gc.gridy=60; p.add(view,gc);

   frame.add(p);

   login=new JButton("Login");
   login.addActionListener(stListener);
   login.setBackground(Color.red);
   login.setForeground(Color.white);
```

```
        login.setBounds(150,330,80,40);

        register=new JButton("Register");
        register.addActionListener(stListener);
        register.setBackground(Color.BLUE);
        register.setForeground(Color.white);
        register.setBounds(250,330,100,40);

        frame.add(login);frame.add(register);

        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setVisible(true);
        frame .setSize(500,500);
    }

    public static void main(String[] args) {
        new StudentForm();
    }
}
package guiForms;

import listeners.LoginListener;
import listeners.StudentListener;

import javax.swing.*;
import java.awt.*;

public class LoginForm {
    public JLabel id,pwd;
    public JTextField tf1,tf2;
    public JButton login,cancel;
    StudentListener stListener;

    public boolean isValidUser=false;

    public LoginForm(StudentListener l)
    {
        this.stListener=l;

        JFrame frame=new JFrame("Login Form");
        frame.setResizable(false);
        frame.getContentPane().setBackground(new Color(2,2,20));
        frame.setLocation(150,150);
        frame.setSize(350,350);
        GridBagLayout g=new GridBagLayout();
        GridBagConstraints gc=new GridBagConstraints();
        frame.setLayout(g);

        id=new JLabel("User ID       :");
        id.setForeground(Color.white);
        pwd=new JLabel("Password    :");
        pwd.setForeground(Color.white);

        tf1=new JTextField(15);
        tf1.setBackground(Color.gray);tf1.setFont(new Font(Font.DIALOG,Font.BOLD,15));
        tf2=new JTextField(15);
```

```
        tf2.setBackground(Color.gray);tf2.setFont(new Font(Font.DIALOG,Font.BOLD,15));

        login=new JButton("Login");
        login.addActionListener(new LoginListener(this,stListener));
        login.setBackground(Color.blue);
        login.setForeground(Color. white);

        Insets i=new Insets(10,10,10,10);
        gc.insets=i;

        gc.gridx=0;gc.gridy=0; frame.add(id,gc);
        gc.gridx=0;gc.gridy=1; frame.add(pwd,gc);

        gc.fill=GridBagConstraints.HORIZONTAL;

        gc.gridx=1;gc.gridy=0; frame.add(tf1,gc);
        gc.gridx=1;gc.gridy=1; frame.add(tf2,gc);

        gc.gridwidth=2;gc.gridx=0;gc.gridy=2; frame.add(login,gc);
        //gc.gridx=1;gc.gridy=2; frame.add(cancel,gc);

        frame.setVisible(true);
        frame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
    }

}


package guiForms;

import listeners.RegisterListener;

import javax.swing.*;
import java.awt.*;

public class RegisterForm {
    public JLabel fname,lname,pwd;
    public JTextField tf1,tf2,tf3;
    public JButton register;

    public RegisterForm()
    {
        JFrame frame=new JFrame("Login Form");
        frame.setResizable(false);
        frame.getContentPane().setBackground(new Color(2,2,20));
        frame.setLocation(150,150);
        frame.setSize(400,400);
        GridBagLayout g=new GridBagLayout();
        GridBagConstraints gc=new GridBagConstraints();
        frame.setLayout(g);

        fname=new JLabel("First Name:");
        fname.setForeground(Color.white);
        lname=new JLabel("Last Name:");
        lname.setForeground(Color.white);
        pwd=new JLabel("Password");
```

```
        pwd.setForeground(Color.white);

        tf1=new JTextField(15);
        tf1.setBackground(Color.gray);tf1.setFont(new Font(Font.DIALOG,Font.BOLD,15));
        tf2=new JTextField(15);
        tf2.setBackground(Color.gray);tf2.setFont(new Font(Font.DIALOG,Font.BOLD,15));
        tf3=new JTextField(15);
        tf3.setBackground(Color.gray);tf3.setFont(new Font(Font.DIALOG,Font.BOLD,15));

        register=new JButton("Register");
        register.addActionListener(new RegisterListener(this));
        register.setBackground(Color.blue);
        register.setForeground(Color.white);

        Insets i=new Insets(10,10,10,10);
        gc.insets=i;
        gc.fill=GridBagConstraints.HORIZONTAL;

        gc.gridx=0;gc.gridy=0; frame.add(fname,gc);
        gc.gridx=0;gc.gridy=1; frame.add(lname,gc);
        gc.gridx=0;gc.gridy=2; frame.add(pwd,gc);

        gc.gridx=1;gc.gridy=0; frame.add(tf1,gc);
        gc.gridx=1;gc.gridy=1; frame.add(tf2,gc);
        gc.gridx=1;gc.gridy=2; frame.add(tf3,gc);

        gc.gridwidth=2;
        gc.gridx=0;gc.gridy=3; frame.add(register,gc);


        frame.setVisible(true);
        frame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
    }

}


package guiForms;

import listeners.AddListener;

import javax.swing.*;
import java.awt.*;

public class AddForm {
    public JLabel name,std,section,dob,address,mobile;
    public JTextField tf1,tf2,tf3,tf4,tf5;
    public JTextArea ta;
    public JButton add;

    public AddForm()
    {
        JFrame frame=new JFrame("Add Student");
        GridBagLayout g=new GridBagLayout();
        GridBagConstraints gc=new GridBagConstraints();
        frame.setLayout(g);
```

```
frame.setResizable(false);
frame.getContentPane().setBackground(new Color(2,2,20));
frame.setLocation(150,150);

Insets i=new Insets(5,5,5,5);
gc.insets=i;

gc.fill=GridBagConstraints.HORIZONTAL;
name=new JLabel("Name",JLabel.LEFT);
name.setForeground(Color.white);
std=new JLabel("Class",JLabel.LEFT);
std.setForeground(Color.white);
section=new JLabel("Section",JLabel.LEFT);
section.setForeground(Color.white);
dob=new JLabel("Date of Birth",JLabel.LEFT);
dob.setForeground(Color.white);
address=new JLabel("Address",JLabel.LEFT);
address.setForeground(Color.white);
mobile=new JLabel("Mobile No.",JLabel.LEFT);
mobile.setForeground(Color.white);

gc.gridx=0;
gc.gridy=0;frame.add(name,gc);
gc.gridy=1;frame.add(std,gc);
gc.gridy=2;frame.add(section,gc);
gc.gridy=3;frame.add(dob,gc);
gc.gridy=4;frame.add(mobile,gc);
gc.gridy=5;frame.add(address,gc);

tf1=new JTextField(15);
tf1.setBackground(Color.gray);tf1.setFont(new Font(Font.DIALOG,Font.BOLD,15));
tf2=new JTextField(15);
tf2.setBackground(Color.gray);tf2.setFont(new Font(Font.DIALOG,Font.BOLD,15));
tf3=new JTextField(15);
tf3.setBackground(Color.gray);tf3.setFont(new Font(Font.DIALOG,Font.BOLD,15));
tf4=new JTextField(15);
tf4.setBackground(Color.gray);tf4.setFont(new Font(Font.DIALOG,Font.BOLD,15));
tf5=new JTextField(15);
tf5.setBackground(Color.gray);tf5.setFont(new Font(Font.DIALOG,Font.BOLD,15));

gc.gridx=1;
gc.gridy=0;frame.add(tf1,gc);
gc.gridy=1;frame.add(tf2,gc);
gc.gridy=2;frame.add(tf3,gc);
gc.gridy=3;frame.add(tf4,gc);
gc.gridy=4;frame.add(tf5,gc);

ta=new JTextArea(5,15);
ta.setBackground(Color.gray);ta.setFont(new Font(Font.DIALOG,Font.BOLD,15));
gc.gridwidth=2;
gc.gridx=0;gc.gridy=6;
frame.add(ta,gc);

add=new JButton("Add Student");
add.addActionListener(new AddListener(this));
add.setBackground(Color.BLUE);add.setForeground(Color.white);
```

```
        gc.gridx=0;gc.gridy=7;
        frame.add(add,gc);

        frame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
        frame.setSize(500,500);
        frame.setVisible(true);

    }

}


package guiForms;

import listeners.RemoveListener;

import javax.swing.*;
import java.awt.*;

public class RemoveForm
{
    public JLabel id;
    public JTextField tf1;
    public JButton remove;

    public RemoveForm()
    {
        JFrame frame=new JFrame("Remove Student");
        GridBagLayout g=new GridBagLayout();
        GridBagConstraints gc=new GridBagConstraints();
        frame.setLayout(g);
        frame.setResizable(false);
        frame.getContentPane().setBackground(new Color(2,2,20));
        frame.setLocation(150,150);

        Insets i=new Insets(10,10,10,10);
        gc.insets=i;
        gc.fill=GridBagConstraints.HORIZONTAL;


        id=new JLabel("Student Id",JLabel.LEFT);
        id.setForeground(Color.white);
        gc.gridx=0;gc.gridy=0;frame.add(id,gc);

        tf1=new JTextField(15);
        tf1.setBackground(Color.gray);tf1.setFont(new Font(Font.DIALOG,Font.BOLD,15));
        gc.gridx=1;gc.gridy=0;frame.add(tf1,gc);

        remove=new JButton("Remove Student");
        remove.addActionListener(new RemoveListener(this));
        remove.setBackground(Color.BLUE);remove.setForeground(Color.white);
        gc.gridx=0;gc.gridy=1;gc.gridwidth=2;frame.add(remove,gc);

        frame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
        frame.setVisible(true);
        frame .setSize(300,300);
```

```
    }

}


package guiForms;

import listeners.UpdateListener;

import javax.swing.*;
import java.awt.*;

public class UpdateForm {

    public JLabel id,cls,sec,db,adr,mob;
    public JTextField tf1,tf2,tf3,tf4,tf5,tf6;
    public JButton updateStd,updateSection,updateDOB,updateAdr,updateMob;

    public UpdateForm()
    {
        JFrame frame=new JFrame("Update Details");
        frame.setLayout(null);
        frame.setResizable(false);
        frame.getContentPane().setBackground(new Color(2,2,20));
        frame.setLocation(150,150);

        id=new JLabel("Student ID");
        id.setForeground(Color.white);
        id.setBounds(80,50,80,30);
        frame.add(id);

        tf1=new JTextField();
        tf1.setBackground(Color.gray);tf1.setFont(new Font(Font.DIALOG,Font.BOLD,15));
        tf1.setBounds(180,50,150,25);
        frame.add(tf1);

        JTabbedPane pane=new JTabbedPane();
        pane.setFont(new Font(Font.DIALOG,Font.BOLD,15));
        pane.setBackground(new Color(10,100,250));
        pane.setForeground(Color.black);
        pane.setBounds(70,100,360,250);
        frame.add(pane);

        Insets i=new Insets(10,10,10,10);

        JPanel std=new JPanel();
        std.setBackground(new Color(2,2,20));
        GridBagLayout g1=new GridBagLayout();
        GridBagConstraints gc1=new GridBagConstraints();
        std.setLayout(g1);
        gc1.insets=i;gc1.fill=GridBagConstraints.HORIZONTAL;
        cls=new JLabel("Class");cls.setForeground(Color.white);gc1.gridx=0;gc1.gridy=0;std.add(cls,gc1);
        tf2=new JTextField(15);tf2.setBackground(Color.gray);tf2.setFont(new
Font(Font.DIALOG,Font.BOLD,15));;gc1.gridx=1;std.add(tf2,gc1);
        updateStd=new JButton("Update
```

```
Class");updateStd.setBackground(Color.blue);updateStd.setForeground(Color.white);gc1.gridx=0;gc1.gridy=1;gc1.gridwidt
h=2;std.add(updateStd,gc1);
        updateStd.addActionListener(new UpdateListener(this));
        pane.addTab("Class",std);

        JPanel section=new JPanel();
        section.setBackground(new Color(2,2,20));
        GridBagLayout g2=new GridBagLayout();
        GridBagConstraints gc2=new GridBagConstraints();
        section.setLayout(g2);
        gc2.insets=i;gc2.fill=GridBagConstraints.HORIZONTAL;
        sec=new JLabel("Section");sec.setForeground(Color.white);gc2.gridx=0;gc2.gridy=0;section.add(sec,gc2);
        tf3=new JTextField(15);tf3.setBackground(Color.gray);tf3.setFont(new
Font(Font.DIALOG,Font.BOLD,15));gc2.gridx=1;section.add(tf3,gc2);
        updateSection=new JButton("Update
Section");updateSection.setBackground(Color.blue);updateSection.setForeground(Color.white);gc2.gridx=0;gc2.gridy=1;
gc2.gridwidth=2;section.add(updateSection,gc2);
        updateSection.addActionListener(new UpdateListener(this));
        pane.addTab("Section",section);

        JPanel dob=new JPanel();
        dob.setBackground(new Color(2,2,20));
        GridBagLayout g3=new GridBagLayout();
        GridBagConstraints gc3=new GridBagConstraints();
        dob.setLayout(g3);
        gc3.insets=i;gc3.fill=GridBagConstraints.HORIZONTAL;
        db=new JLabel("Date of Birth");db.setForeground(Color.white);gc3.gridx=0;gc3.gridy=0;dob.add(db,gc3);
        tf4=new JTextField(15);tf4.setBackground(Color.gray);tf4.setFont(new
Font(Font.DIALOG,Font.BOLD,15));gc3.gridx=1;dob.add(tf4,gc3);
        updateDOB=new JButton("Update
DOB");updateDOB.setBackground(Color.blue);updateDOB.setForeground(Color.white);gc3.gridx=0;gc3.gridy=1;gc3.gridwi
dth=2;dob.add(updateDOB,gc3);
        updateDOB.addActionListener(new UpdateListener(this));
        pane.addTab("DOB",dob);


        JPanel address=new JPanel();
        address.setBackground(new Color(2,2,20));
        GridBagLayout g4=new GridBagLayout();
        GridBagConstraints gc4=new GridBagConstraints();
        address.setLayout(g4);
        gc4.insets=i;gc4.fill=GridBagConstraints.HORIZONTAL;
        adr=new JLabel("Address");adr.setForeground(Color.white);gc4.gridx=0;gc4.gridy=0;address.add(adr,gc4);
        tf5=new JTextField(15);tf5.setBackground(Color.gray);tf5.setFont(new
Font(Font.DIALOG,Font.BOLD,15));gc4.gridx=1;address.add(tf5,gc4);
        updateAdr=new JButton("Update
Address");updateAdr.setBackground(Color.blue);updateAdr.setForeground(Color.white);gc4.gridx=0;gc4.gridy=1;gc4.gri
dwidth=2;address.add(updateAdr,gc4);
        updateAdr.addActionListener(new UpdateListener(this));
        pane.addTab("Address",address);

        JPanel mobile=new JPanel();
        mobile.setBackground(new Color(2,2,20));
        GridBagLayout g5=new GridBagLayout();
        GridBagConstraints gc5=new GridBagConstraints();
        mobile.setLayout(g5);
```

```
        gc5.insets=i;gc5.fill=GridBagConstraints.HORIZONTAL;
        mob=new JLabel("Mobile No.");mob.setForeground(Color.white);gc5.gridx=0;gc5.gridy=0;mobile.add(mob,gc5);
        tf6=new JTextField(15);tf6.setBackground(Color.gray);tf6.setFont(new
Font(Font.DIALOG,Font.BOLD,15));gc5.gridx=1;mobile.add(tf6,gc5);
        updateMob=new JButton("Update
Contact");updateMob.setBackground(Color.blue);updateMob.setForeground(Color.white);gc5.gridx=0;gc5.gridy=1;gc5.gri
dwidth=2;mobile.add(updateMob,gc5);
        updateMob.addActionListener(new UpdateListener(this));
        pane.addTab("Mobile",mobile);


        frame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
        frame.setVisible(true);
        frame.setSize(500,450);
    }

}


package guiForms;

import listeners.ViewListener;

import javax.swing.*;
import javax.swing.table.DefaultTableModel;
import java.awt.*;

public class ViewForm {
    public JLabel id;
    public JTextField tf1;
    public JButton view,viewAll;
    public DefaultTableModel model;
    public JTable table;

    public ViewForm()
    {
        JFrame frame=new JFrame("View Details");
        frame.setLayout(null);
        frame.setResizable(false);
        frame.getContentPane().setBackground(new Color(2,2,20));
        frame.setLocation(150,150);

        GridBagLayout g=new GridBagLayout();
        GridBagConstraints gc=new GridBagConstraints();
        JPanel p=new JPanel();
        p.setBounds(50,50,400,100);
        p.setLayout(g);p.setBackground(new Color(2,2,20));
        frame.add(p);

        Insets i=new Insets(10,10,10,10);
        gc.insets=i;gc.fill=GridBagConstraints.HORIZONTAL;

        id=new JLabel("Student id",JLabel.LEFT);
        id.setForeground(Color.white);
        gc.gridx=0;gc.gridy=0;p.add(id,gc);
```

```
        tf1=new JTextField(15);
        tf1.setBackground(Color.gray);tf1.setFont(new Font(Font.DIALOG,Font.BOLD,15));
        gc.gridx=1;p.add(tf1,gc);

        view=new JButton("View Details");
        view.addActionListener(new ViewListener(this));
        view.setBackground(Color.BLUE);view.setForeground(Color.white);
        gc.gridx=0;gc.gridy=1;p.add(view,gc);

        viewAll=new JButton("View all Details");
        viewAll.addActionListener(new ViewListener(this));
        gc.gridx=1;gc.gridy=1;p.add(viewAll,gc);


        String[] cols={"ID","Name","Class","Section","DOB","Mobile","Permanent Address"};
        model=new DefaultTableModel();
        model.setColumnIdentifiers(cols);
        table=new JTable(model);
        table.setGridColor(Color.black);
        table.setForeground(Color.white);
        table.setBackground(Color.darkGray);
        JScrollPane pane=new JScrollPane(table);
        pane.setBounds(50,200,400,200);
        frame.add(pane);

        frame.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);
        frame.setVisible(true);
        frame .setSize(500,500);
    }

}



package listeners;

import guiForms.*;

import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class StudentListener implements ActionListener {
    boolean isLoggedIn=false;

    @Override
    public void actionPerformed(ActionEvent e) {
        String action=e.getActionCommand();
        if(action.equals("Login"))
        {
            new LoginForm(this);
        }
        else if(action.equals("Register"))
        {
            new RegisterForm();
        }
        else if(action.equals("Add Student"))
```

```
        {
            if(isLoggedIn==false)
                new LoginForm(this);
            else
                new AddForm();
        }
        else if(action.equals("Update Details"))
        {
            if(isLoggedIn==false)
                new LoginForm(this);
            else
                new UpdateForm();
        }
        else if(action.equals("Remove Student"))
        {
            if(isLoggedIn==false)
                new LoginForm(this);
            else
                new RemoveForm();
        }
        else
        {
            if(isLoggedIn==false)
                new LoginForm(this);
            else
                new ViewForm();
        }
    }
}


package listeners;

import beans.LoginBean;
import dao.LoginDao;
import guiForms.LoginForm;

import javax.swing.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class LoginListener implements ActionListener {
    LoginForm loginForm;
    StudentListener stListener;
    boolean isValidUser;
    public LoginListener(LoginForm f,StudentListener l)
    {
        this.loginForm=f;
        this.stListener=l;
    }

    @Override
    public void actionPerformed(ActionEvent e) {
        String id=loginForm.tf1.getText();
        String pwd=loginForm.tf2.getText();
```

```
        loginForm.tf1.setText("");
        loginForm.tf2.setText("");

        if(pwd.equals("")||id.equals(""))
        {
            JOptionPane.showMessageDialog(null,"Please fill the details!");
        }
        else
        {
            LoginBean l=new LoginBean(Integer.parseInt(id),pwd);
            isValidUser= LoginDao.isValidUser(l);

            if(isValidUser)
            {
                JOptionPane.showMessageDialog(null,"You are successfully logged in!");
                stListener.isLoggedIn=true;
            }
            else
                JOptionPane.showMessageDialog(null,"Please enter valid details!");
        }
    }
}


package listeners;

import beans.RegisterBean;
import dao.RegisterDao;
import guiForms.RegisterForm;

import javax.swing.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class RegisterListener implements ActionListener {
    public RegisterForm registerForm;

    public RegisterListener(RegisterForm f)
    {
        this.registerForm =f;
    }
    public void actionPerformed(ActionEvent e)
    {
        String fname=registerForm.tf1.getText();
        String lname=registerForm.tf2.getText();
        String pwd=registerForm.tf3.getText();

        registerForm.tf1.setText("");
        registerForm.tf2.setText("");
        registerForm.tf3.setText("");

        if(fname.equals("")||lname.equals("")||pwd.equals(""))
            JOptionPane.showMessageDialog(null,"Please enter the details!");
        else
        {
            RegisterBean b=new RegisterBean(fname,lname,pwd);
```

```
            int id= RegisterDao.registerUser(b);
            if(id!=-1)
            {
               JOptionPane.showMessageDialog(null,"Your id is : "+id);
            }
            else
               JOptionPane.showMessageDialog(null,"Registration failed! Try Again.");
         }
      }
}
```

```
package listeners;

import beans.AddBean;
import dao.AddDao;
import guiForms.AddForm;

import javax.swing.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class AddListener implements ActionListener {
   AddForm addForm;
   public AddListener(AddForm f)
   {
      addForm=f;
   }
   @Override
   public void actionPerformed(ActionEvent e) {
      String name=addForm.tf1.getText();
      String std=addForm.tf2.getText();
      String section=addForm.tf3.getText();
      String dob=addForm.tf4.getText();
      String mobile=addForm.tf5.getText();
      String address=addForm.ta.getText();

      addForm.tf1.setText("");
      addForm.tf2.setText("");
      addForm.tf3.setText("");
      addForm.tf4.setText("");
      addForm.tf5.setText("");
      addForm.ta.setText("");


      if(name.equals("")||std.equals("")||section.equals("")||dob.equals("")||mobile.equals("")||address.equals(""))
         JOptionPane.showMessageDialog(null,"Please fill the details");
      else
      {
         AddBean b=new AddBean(name,Integer.parseInt(std),section,dob,mobile,address);
         int id= AddDao.addStudent(b);
         if(id!=-1)
         {
            JOptionPane.showMessageDialog(null,"Details are submitted successfully!\nStudent id : "+id);
         }
         else
```

```
            JOptionPane.showMessageDialog(null,"Submission Failed! Try Again.");
        }
    }
}
```

```
package listeners;

import dao.RemoveDao;
import guiForms.RemoveForm;

import javax.swing.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class RemoveListener implements ActionListener {
    RemoveForm removeForm;
    public RemoveListener(RemoveForm f)
    {
        removeForm=f;
    }
    @Override
    public void actionPerformed(ActionEvent e) {
        String id=removeForm.tf1.getText();

        removeForm.tf1.setText("");

        if(id.equals(""))
            JOptionPane.showMessageDialog(null,"Please enter the id!");
        else
        {
            boolean isRemoved= RemoveDao.removeStudent(Integer.parseInt(id));
            if(isRemoved)
                JOptionPane.showMessageDialog(null,"Removed Successfully!");
            else
                JOptionPane.showMessageDialog(null,"Failed! Try again.");
        }
    }
}
```

```
package listeners;

import dao.UpdateDao;
import guiForms.UpdateForm;

import javax.swing.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class UpdateListener implements ActionListener {
    UpdateForm updateForm;
    public UpdateListener(UpdateForm f)
    {
        updateForm=f;
    }
```

```
    @Override
    public void actionPerformed(ActionEvent e) {
        String id=updateForm.tf1.getText();
        updateForm.tf1.setText("");

        String newValue="";
        String action=e.getActionCommand();
        boolean isUpdated=false;
        if(action.equals("Update Class"))
        {
            newValue=updateForm.tf2.getText();
            updateForm.tf2.setText("");
            isUpdated= UpdateDao.updateClass(Integer.parseInt(id),newValue);
        }
        else if(action.equals("Update Section"))
        {
            newValue=updateForm.tf3.getText();
            updateForm.tf3.setText("");
            isUpdated= UpdateDao.updateSection(Integer.parseInt(id),newValue);
        }
        else if(action.equals("Update DOB"))
        {
            newValue=updateForm.tf4.getText();
            updateForm.tf4.setText("");
            isUpdated= UpdateDao.updateDOB(Integer.parseInt(id),newValue);
        }
        else if(action.equals("Update Address"))
        {
            newValue=updateForm.tf5.getText();
            updateForm.tf5.setText("");
            isUpdated= UpdateDao.updateAddress(Integer.parseInt(id),newValue);
        }
        else
        {
            newValue=updateForm.tf6.getText();
            updateForm.tf6.setText("");
            isUpdated= UpdateDao.updateContact(Integer.parseInt(id),newValue);
        }

        if(isUpdated)
        {
            JOptionPane.showMessageDialog(null,"Details are updated!");
        }
        else
            JOptionPane.showMessageDialog(null,"Updation Failed! Try Again.");
    }
}


package listeners;

import dao.ViewDao;
import guiForms.ViewForm;

import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
```

```
public class ViewListener implements ActionListener {
   ViewForm viewForm;
   public ViewListener(ViewForm f)
   {
      viewForm=f;
   }
   @Override
   public void actionPerformed(ActionEvent e) {
      String id=viewForm.tf1.getText();
      viewForm.tf1.setText("");
      String action=e.getActionCommand();

      viewForm.model.setRowCount(0);
      if(action.equals("View all Details"))
         ViewDao.viewAllDetails(viewForm);
      else
         ViewDao.viewDetails(viewForm,Integer.parseInt(id));
   }
}
```

```
package dao;

import beans.LoginBean;
import connection.ConnectionProvider;

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;

public class LoginDao {

   public static boolean isValidUser(LoginBean b)
   {
      boolean isValid=false;
      Connection con=null;
      try
      {
         con= ConnectionProvider.provideConnection();
         PreparedStatement p=con.prepareStatement("select * from user where id=? and pwd=?");
         p.setInt(1,b.id);
         p.setString(2,b.pwd);
         ResultSet rs=p.executeQuery();

         if(rs.next())
            isValid=true;
      }
      catch(SQLException e)
      {
         System.out.println(e);
      }
      finally {
         try
         {
            con.close();
```

```
        }
        catch (SQLException e)
        {
           System.out.println(e);
        }
     }
     return isValid;
  }
}
```

```
package dao;

import beans.RegisterBean;
import connection.ConnectionProvider;

import java.sql.*;

public class RegisterDao {
   public static int registerUser(RegisterBean b)
   {
      int id=-1;
      Connection con=null;

      try
      {
         con= ConnectionProvider.provideConnection();
         PreparedStatement p=con.prepareStatement("insert into user(lname,fname,pwd) values(?,?,?);");
         p.setString(1,b.fname);
         p.setString(2,b.lname);
         p.setString(3,b.pwd);
         int r=p.executeUpdate();

         if(r>0)
         {
            Statement s=con.createStatement();
            ResultSet rs=s.executeQuery("select max(id) from user;");
            rs.next();
            id=rs.getInt(1);
         }
      }
      catch(SQLException e)
      {
         System.out.println(e);
      }
      finally
      {
         try
         {
            con.close();
         }
         catch (SQLException e)
         {
            System.out.println(e);
         }
      }
```

```
        return id;
    }
}
```

```
package dao;

import beans.AddBean;
import connection.ConnectionProvider;

import java.sql.*;

public class AddDao {
    public static int addStudent(AddBean b)
    {
        int id=-1;
        Connection con=null;
        try
        {
            con= ConnectionProvider.provideConnection();
            PreparedStatement p=con.prepareStatement("insert into student(name,class,section,dob,address,mobile)
values(?,?,?,?,?,?);");
            p.setString(1,b.name);
            p.setInt(2,b.std);
            p.setString(3,b.section);
            p.setDate(4,Date.valueOf(b.dob));
            p.setString(5,b.address);
            p.setString(6,b.mobile);

            int r=p.executeUpdate();

            if(r>0)
            {
                Statement s=con.createStatement();
                ResultSet rs=s.executeQuery("select max(id) from student");
                rs.next();
                id=rs.getInt(1);
            }
        }
        catch(SQLException e)
        {
            System.out.println(e);
        }
        finally {
            try
            {
                con.close();
            }
            catch (SQLException e) {
                System.out.println(e);
            }
        }

        return id;
    }
}
```

```
package dao;

import connection.ConnectionProvider;

import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.SQLException;

public class RemoveDao {

public static boolean removeStudent(int id)
{
   boolean isRemoved =false;
   Connection con=null;
   try
   {
      con= ConnectionProvider.provideConnection();
      PreparedStatement p=con.prepareStatement("delete from student where id=?");
      p.setInt(1,id);
      int r=p.executeUpdate();
      if(r>0)
         isRemoved=true;
   }
   catch(SQLException e)
   {
      System.out.println(e);
   }
   finally {
      try{con.close();}
      catch(SQLException e) {
         System.out.println(e);
      }
   }

   return isRemoved;
}
}

package dao;

import connection.ConnectionProvider;

import java.sql.Connection;
import java.sql.Date;
import java.sql.PreparedStatement;
import java.sql.SQLException;

public class UpdateDao {
   public static boolean updateClass(int id,String newValue)
   {
      boolean isClassUpdated=false;
      Connection con=null;

      try
```

```
    {
        con= ConnectionProvider.provideConnection();
        PreparedStatement p=con.prepareStatement("update student set class=? where id=?;");
        p.setString(1,newValue);
        p.setInt(2,id);
        int r=p.executeUpdate();
        if(r>0)isClassUpdated=true;
    }
    catch (SQLException e)
    {
        System.out.println(e);
    }
    finally {
        try{con.close();}
        catch (SQLException e) {
            System.out.println(e);
        }
    }

    return isClassUpdated;
}

public static boolean updateSection(int id,String newValue)
{
    boolean isSectionUpdated=false;
    Connection con=null;

    try
    {
        con= ConnectionProvider.provideConnection();
        PreparedStatement p=con.prepareStatement("update student set section=? where id=?;");
        p.setString(1,newValue);
        p.setInt(2,id);
        int r=p.executeUpdate();
        if(r>0)isSectionUpdated=true;
    }
    catch (SQLException e)
    {
        System.out.println(e);
    }
    finally {
        try{con.close();}
        catch (SQLException e) {
            System.out.println(e);
        }
    }

    return isSectionUpdated;
}
public static boolean updateDOB(int id,String newValue)
{
    boolean isDOBUpdated=false;
    Connection con=null;

    try
    {
```

```
        con= ConnectionProvider.provideConnection();
        PreparedStatement p=con.prepareStatement("update student set dob=? where id=?;");
        p.setDate(1,Date.valueOf(newValue));
        p.setInt(2,id);
        int r=p.executeUpdate();
        if(r>0)isDOBUpdated=true;
    }
    catch (SQLException e)
    {
        System.out.println(e);
    }
    finally {
        try{con.close();}
        catch (SQLException e) {
            System.out.println(e);
        }
    }

    return isDOBUpdated;
}
public static boolean updateAddress(int id,String newValue)
{
    boolean isAddressUpdated=false;
    Connection con=null;

    try
    {
        con= ConnectionProvider.provideConnection();
        PreparedStatement p=con.prepareStatement("update student set address=? where id=?;");
        p.setString(1,newValue);
        p.setInt(2,id);
        int r=p.executeUpdate();
        if(r>0)isAddressUpdated=true;
    }
    catch (SQLException e)
    {
        System.out.println(e);
    }
    finally {
        try{con.close();}
        catch (SQLException e) {
            System.out.println(e);
        }
    }

    return isAddressUpdated;
}
public static boolean updateContact(int id,String newValue)
{
    boolean isContactUpdated=false;
    Connection con=null;

    try
    {
        con= ConnectionProvider.provideConnection();
        PreparedStatement p=con.prepareStatement("update student set mobile=? where id=?;");
```

```
            p.setString(1,newValue);
            p.setInt(2,id);
            int r=p.executeUpdate();
            if(r>0)isContactUpdated=true;
        }
        catch (SQLException e)
        {
            System.out.println(e);
        }
        finally {
            try{con.close();}
            catch (SQLException e) {
                System.out.println(e);
            }
        }

        return isContactUpdated;
    }

}


package dao;

import connection.ConnectionProvider;
import guiForms.ViewForm;

import java.sql.*;

public class ViewDao {
    public static void viewAllDetails(ViewForm f)
    {
        String[] data=new String[7];
        Connection con=null;
        try
        {
            con= ConnectionProvider.provideConnection();
            Statement s=con.createStatement();
            ResultSet rs=s.executeQuery("select * from student;");

            while(rs.next())
            {
                data[0]=rs.getInt(1)+"";
                data[1]=rs.getString(2);
                data[2]=rs.getString(3);
                data[3]=rs.getString(4);
                data[4]=rs.getString(5);
                data[5]=rs.getString(6);
                data[6]=rs.getString(7);

                f.model.addRow(data);
            }
        }
        catch(SQLException e)
        {
            System.out.println(e);
```

```
        }
        finally {
            try{con.close();}
            catch (SQLException e){System.out.println(e);}
        }
    }

    public static void viewDetails(ViewForm f,int id)
    {
        String[] data=new String[7];
        Connection con=null;
        try
        {
            con= ConnectionProvider.provideConnection();
            PreparedStatement p=con.prepareStatement("select * from student where id=?;");
            p.setInt(1,id);
            ResultSet rs=p.executeQuery();
            rs.next();
            data[0]=rs.getInt(1)+"";
            data[1]=rs.getString(2);
            data[2]=rs.getString(3);
            data[3]=rs.getString(4);
            data[4]=rs.getString(5);
            data[5]=rs.getString(6);
            data[6]=rs.getString(7);
            f.model.addRow(data);
        }
        catch(SQLException e)
        {
            System.out.println(e);
        }
        finally {
            try{con.close();}
            catch (SQLException e){System.out.println(e);}
        }
    }
}


package beans;

public class LoginBean {
    public int id;
    public String pwd;
    public LoginBean(int id,String pwd)
    {
        this.id=id;
        this.pwd=pwd;
    }
}


package beans;

public class RegisterBean {
    public String fname;
    public String lname;
```

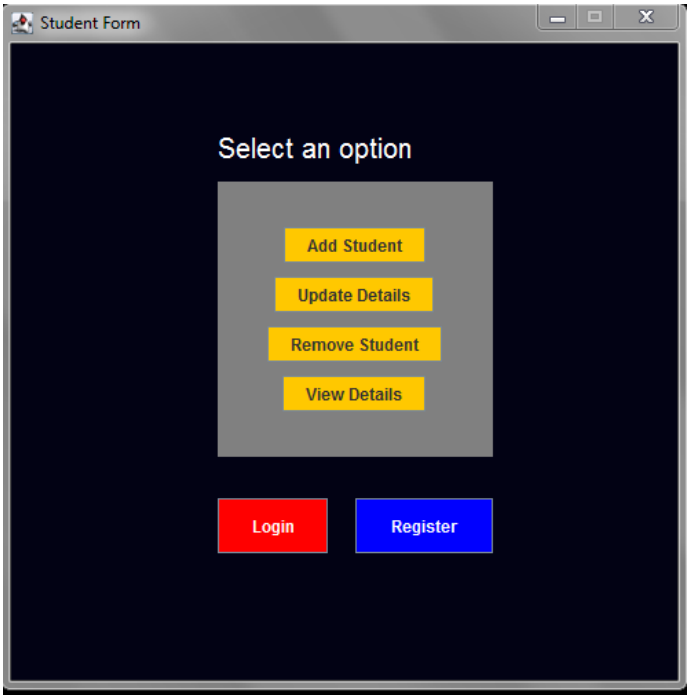
```
    public String pwd;

    public RegisterBean(String fname,String lname,String pwd)
    {
        this.fname=fname;
        this.lname=lname;
        this.pwd=pwd;
    }
}
```

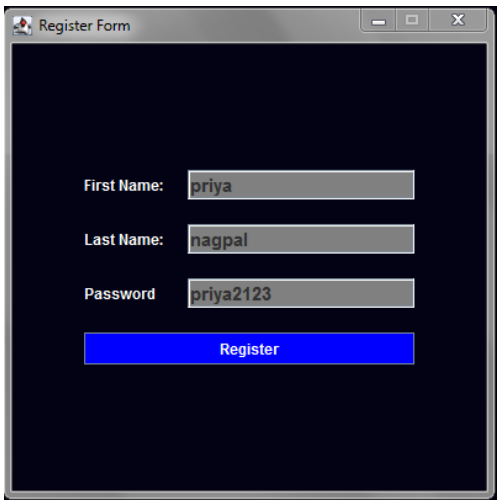
```
package beans;

public class AddBean {
    public String name;
    public int std;
    public String section;
    public String dob;
    public String mobile;
    public String address;

    public AddBean(String name, int std, String section, String dob, String mobile,String address)
    {
        this.name=name;
        this.std=std;
        this.section=section;
        this.dob=dob;
        this.mobile=mobile;
        this.address=address;
    }
}
```



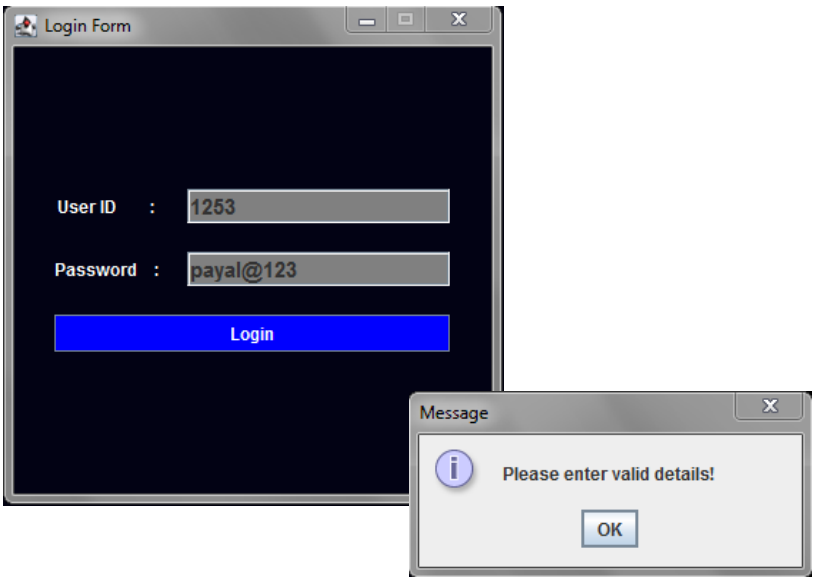
Register as user :

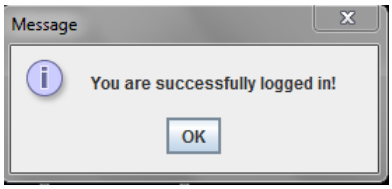



```
mysql> select * from user;
+-------+---------+--------+-------------+
| id    | fname   | lname  | pwd         |
+-------+---------+--------+-------------+
| 12345 | payal   | rathee | payal@123   |
| 12347 | kumar   | naman  | naman@123   |
| 12348 | ahlawat | priya  | priya@123   |
| 12349 | dahiya  | seema  | seema@123   |
| 12350 | kumar   | manish | manish@123  |
| 12351 | rathee  | bhawna | bhanu@123   |
| 12352 | ahlawat | diksha | diksha@123  |
+-------+---------+--------+-------------+
```

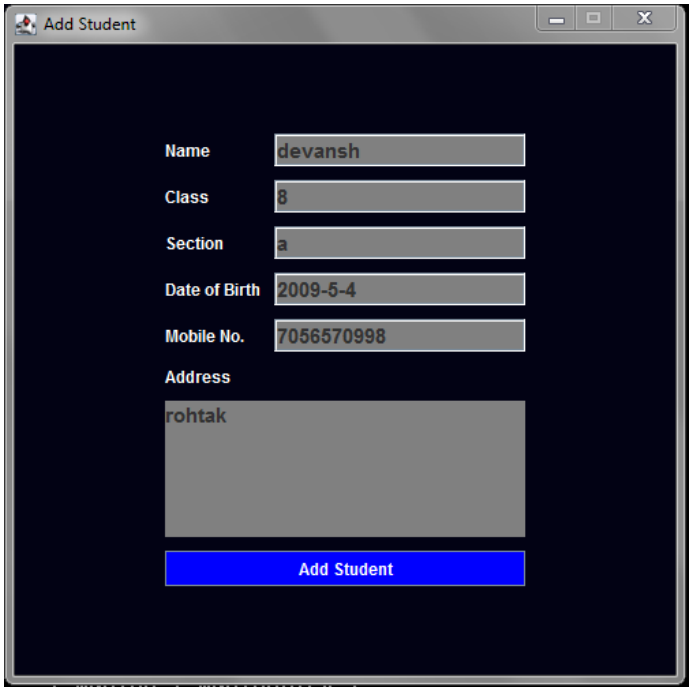
```
mysql> select * from user;
+-------+---------+--------+-------------+
| id    | fname   | lname  | pwd         |
+-------+---------+--------+-------------+
| 12345 | payal   | rathee | payal@123   |
| 12347 | kumar   | naman  | naman@123   |
| 12348 | ahlawat | priya  | priya@123   |
| 12349 | dahiya  | seema  | seema@123   |
| 12350 | kumar   | manish | manish@123  |
| 12351 | rathee  | bhawna | bhanu@123   |
| 12352 | ahlawat | diksha | diksha@123  |
| 12353 | nagpal  | priya  | priya2123   |
+-------+---------+--------+-------------+
```

Login :





Add Student :



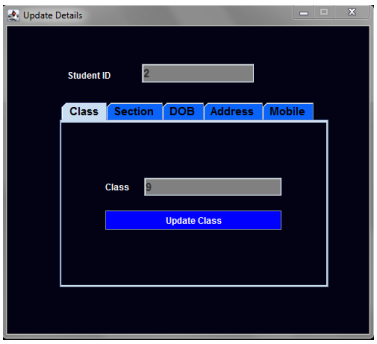
Before :

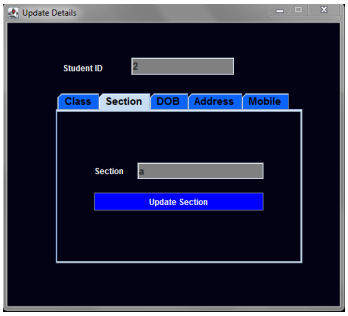
```
mysql> select * from student;
+----+--------+-------+---------+------------+---------+------------+
| id | name   | class | section | dob        | address | mobile     |
+----+--------+-------+---------+------------+---------+------------+
|  2 | naman  |     8 | b       | 2008-03-04 | ambala  | 9783462109 |
|  8 | riya   |     8 | b       | 2008-03-05 | ambala  | 9837483282 |
+----+--------+-------+---------+------------+---------+------------+
```

After :

```
mysql> select * from student;
+----+---------+-------+---------+------------+---------+------------+
| id | name    | class | section | dob        | address | mobile     |
+----+---------+-------+---------+------------+---------+------------+
|  2 | naman   |     8 | b       | 2008-03-04 | ambala  | 9783462109 |
|  8 | riya    |     8 | b       | 2008-03-05 | ambala  | 9837483282 |
|  9 | devansh |     8 | a       | 2009-05-04 | rohtak  | 7056570998 |
+----+---------+-------+---------+------------+---------+------------+
```

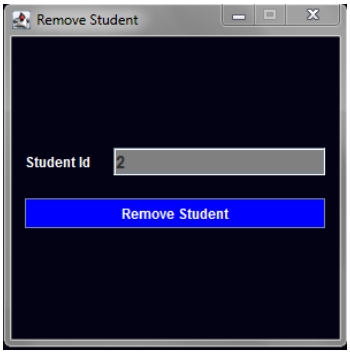
Update Details :

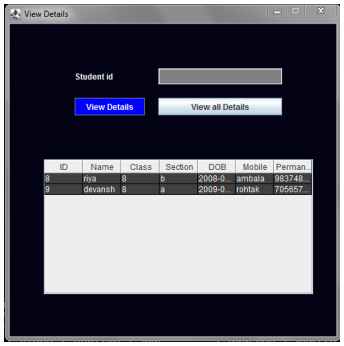





```
mysql> select * from student;
+----+---------+-------+---------+------------+---------+------------+
| id | name    | class | section | dob        | address | mobile     |
+----+---------+-------+---------+------------+---------+------------+
|  2 | naman   |     9 | a       | 2008-03-04 | ambala  | 9783462109 |
|  8 | riya    |     8 | b       | 2008-03-05 | ambala  | 9837483282 |
|  9 | devansh |     8 | a       | 2009-05-04 | rohtak  | 7056570998 |
+----+---------+-------+---------+------------+---------+------------+
```

Remove Student :




```
mysql> select * from student;
+----+---------+-------+---------+------------+---------+------------+
| id | name    | class | section | dob        | address | mobile     |
+----+---------+-------+---------+------------+---------+------------+
|  8 | riya    |     8 | b       | 2008-03-05 | ambala  | 9837483282 |
|  9 | devansh |     8 | a       | 2009-05-04 | rohtak  | 7056570998 |
+----+---------+-------+---------+------------+---------+------------+
```

View Details :