AMITY SCHOOL OF ENGINEERING & TECHNOLOGY

AMITY UNIVERSITY CAMPUS, SECTOR-125, NOIDA-201303

Java Mini Project

COURSE CODE: IT 201

Submitted to:

Mrs.Anju Mishra

Submitted by:

1) Abhishek Shekhawat (A2305319018)
2) Shikhar Trivedi (A2305319004)
3) Shobhit Harit(A2305319028)
4) Jatin Panwar (A2305319005)
5) Amit Koyal(A2305319002)
6)Naman Awasthi(A2305319131)

Problem Statement:

The College library has to be automated. The system has to perform multiple stakeholders and would have have different logins. The system should be able to process Book Issue, Process Book Return, Search Book, Catalogue a new Book. Use file systems for backend.

Software Used: Java IDE, MySQL

Theory:-

Library Automation software, also known as an automated library management system is a software that has been developed to handle basic housekeeping functions of a library. The function may be either single function or integrated and realizing the important role of library management system will play in planning and implementing library automation projects.

It is an enterprise resource planning system for a library used to track items owned, orders made, bills paid, and patrons who have borrowed and usually comprised of a relations database, software to act on that database, and two graphical user interfaces.

The basic purpose of this project to make a college library database, which can be used to perform multiple stakeholders and would have different logins. The system should be able to process Book Issue, Process Book Return, Search Book, Catalogue a new Book. We have used java for frontend programming and for backend we have used mysql.

Code:

```
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.*;
import java.text.DateFormat;
import java.text.ParseException;
import java.text.SimpleDateFormat;
import java.util.ArrayList;
import java.util.Date;
import java.util.Locale;
import java.util.concurrent.TimeUnit;

import javax.swing.*;
import net.proteanit.sql.DbUtils;

public class main {

   public static class ex{
      public static int days=0;
         }

   public static void main(String[] args) {

      login();
      //create();
   }
public static void login() {

   JFrame f=new JFrame("Login");//creating instance of JFrame
```

```
  JLabel l1,l2;
  l1=new JLabel("Username");  //Create label Username
  l1.setBounds(30,15, 100,30); //x axis, y axis, width, height

  l2=new JLabel("Password");  //Create label Password
  l2.setBounds(30,50, 100,30);

  JTextField F_user = new JTextField(); //Create text field for username
  F_user.setBounds(110, 15, 200, 30);

  JPasswordField F_pass=new JPasswordField(); //Create text field for password
  F_pass.setBounds(110, 50, 200, 30);

  JButton login_but=new JButton("Login");//creating instance of JButton for Login Button
  login_but.setBounds(130,90,80,25);//Dimensions for button
  login_but.addActionListener(new ActionListener() {  //Perform action

    public void actionPerformed(ActionEvent e){

    String username = F_user.getText(); //Store username entered by the user in the variable "username"
    String password = F_pass.getText(); //Store password entered by the user in the variable "password"

    if(username.equals("")) //If username is null
    {
      JOptionPane.showMessageDialog(null,"Please enter username"); //Display dialog box with the
message
    }
    else if(password.equals("")) //If password is null
    {
      JOptionPane.showMessageDialog(null,"Please enter password"); //Display dialog box with the
message
    }
    else { //If both the fields are present then to login the user, check wether the user exists already
      //System.out.println("Login connect");
      Connection connection=connect();  //Connect to the database
      try
      {
      Statement stmt = connection.createStatement();
       stmt.executeUpdate("USE LIBRARY"); //Use the database with the name "Library"
       String st = ("SELECT * FROM USERS WHERE USERNAME='"+username+"' AND
PASSWORD='"+password+"'"); //Retreive username and passwords from users
       ResultSet rs = stmt.executeQuery(st); //Execute query
       if(rs.next()==false) { //Move pointer below
         System.out.print("No user");
         JOptionPane.showMessageDialog(null,"Wrong Username/Password!"); //Display Message

       }
       else {
         f.dispose();
        rs.beforeFirst();  //Move the pointer above
        while(rs.next())
        {
         String admin = rs.getString("ADMIN"); //user is admin
         //System.out.println(admin);
         String UID = rs.getString("UID"); //Get user ID of the user
         if(admin.equals("1")) { //If boolean value 1
           admin_menu(); //redirect to admin menu
         }
         else{
           user_menu(UID); //redirect to user menu for that user ID
         }
       }
       }
      }
      catch (Exception ex) {
         ex.printStackTrace();
    }
    }
  }
  });


  f.add(F_pass); //add password
  f.add(login_but);//adding button in JFrame
  f.add(F_user);  //add user
  f.add(l1);  // add label1 i.e. for username
  f.add(l2); // add label2 i.e. for password

  f.setSize(400,180);//400 width and 500 height
  f.setLayout(null);//using no layout managers
  f.setVisible(true);//making the frame visible
  f.setLocationRelativeTo(null);

}
public static Connection connect()
{
try {
    Class.forName("com.mysql.cj.jdbc.Driver");
    //System.out.println("Loaded driver");
    Connection con =
DriverManager.getConnection("jdbc:mysql://localhost/mysql?user=root&password=asn");
    //System.out.println("Connected to MySQL");
    return con;
 }
 catch (Exception ex) {
    ex.printStackTrace();
 }
return null;
}
public static void create() {
  try {
  Connection connection=connect();
  ResultSet resultSet = connection.getMetaData().getCatalogs();
  //iterate each catalog in the ResultSet
    while (resultSet.next()) {
     // Get the database name, which is at position 1
     String databaseName = resultSet.getString(1);
     if(databaseName.equals("library")) {
       //System.out.print("yes");
       Statement stmt = connection.createStatement();
       //Drop database if it pre-exists to reset the complete database
       String sql = "DROP DATABASE library";
       stmt.executeUpdate(sql);
     }
    }
     Statement stmt = connection.createStatement();

     String sql = "CREATE DATABASE LIBRARY"; //Create Database
     stmt.executeUpdate(sql);
     stmt.executeUpdate("USE LIBRARY"); //Use Database
     //Create Users Table
     String sql1 = "CREATE TABLE USERS(UID INT NOT NULL AUTO_INCREMENT PRIMARY
KEY, USERNAME VARCHAR(30), PASSWORD VARCHAR(30), ADMIN BOOLEAN)";
     stmt.executeUpdate(sql1);
     //Insert into users table
     stmt.executeUpdate("INSERT INTO USERS(USERNAME, PASSWORD, ADMIN)
VALUES('admin','admin',TRUE)");
     //Create Books table
     stmt.executeUpdate("CREATE TABLE BOOKS(BID INT NOT NULL AUTO_INCREMENT
PRIMARY KEY, BNAME VARCHAR(50), GENRE VARCHAR(20), PRICE INT)");
     //Create Issued Table
     stmt.executeUpdate("CREATE TABLE ISSUED(IID INT NOT NULL AUTO_INCREMENT
PRIMARY KEY, UID INT, BID INT, ISSUED_DATE VARCHAR(20), RETURN_DATE VARCHAR(20),
PERIOD INT, FINE INT)");
     //Insert into books table
     stmt.executeUpdate("INSERT INTO BOOKS(BNAME, GENRE, PRICE) VALUES ('War and Peace',
'Mystery', 200),  ('The Guest Book', 'Fiction', 300), ('The Perfect Murder','Mystery', 150), ('Accidental
Presidents', 'Biography', 250), ('The Wicked King','Fiction', 350)");

  resultSet.close();
  }
   catch (Exception ex) {
       ex.printStackTrace();
}
}
public static void user_menu(String UID) {


  JFrame f=new JFrame("User Functions"); //Give dialog box name as User functions
  //f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); //Exit user menu on closing the dialog box
  JButton view_but=new JButton("View Books");//creating instance of JButton
  view_but.setBounds(20,20,120,25);//x axis, y axis, width, height
  view_but.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e){

      JFrame f = new JFrame("Books Available"); //View books stored in database
      //f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);



      Connection connection = connect();
      String sql="select * from BOOKS"; //Retreive data from database
      try {
```

```
          Statement stmt = connection.createStatement(); //connect to database
           stmt.executeUpdate("USE LIBRARY"); // use librabry
          stmt=connection.createStatement();
          ResultSet rs=stmt.executeQuery(sql);
          JTable book_list= new JTable(); //show data in table format
          book_list.setModel(DbUtils.resultSetToTableModel(rs));

          JScrollPane scrollPane = new JScrollPane(book_list); //enable scroll bar

          f.add(scrollPane); //add scroll bar
          f.setSize(800, 400); //set dimensions of view books frame
          f.setVisible(true);
          f.setLocationRelativeTo(null);
        } catch (SQLException e1) {
          // TODO Auto-generated catch block
           JOptionPane.showMessageDialog(null, e1);
        }

  }
  }
  );

  JButton my_book=new JButton("My Books");//creating instance of JButton
  my_book.setBounds(150,20,120,25);//x axis, y axis, width, height
  my_book.addActionListener(new ActionListener() { //Perform action
      public void actionPerformed(ActionEvent e){


        JFrame f = new JFrame("My Books"); //View books issued by user
        //f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        int UID_int = Integer.parseInt(UID); //Pass user ID

        //.iid,issued.uid,issued.bid,issued.issued_date,issued.return_date,issued,
        Connection connection = connect(); //connect to database
        //retrieve data
        String sql="select distinct issued.*,books.bname,books.genre,books.price from issued,books " + "where
((issued.uid=" + UID_int + ") and (books.bid in (select bid from issued where issued.uid="+UID_int+")))
group by iid";
        String sql1 = "select bid from issued where uid="+UID_int;
        try {
          Statement stmt = connection.createStatement();
          //use database
           stmt.executeUpdate("USE LIBRARY");
          stmt=connection.createStatement();
          //store in array
          ArrayList books_list = new ArrayList();



          ResultSet rs=stmt.executeQuery(sql);
          JTable book_list= new JTable(); //store data in table format
          book_list.setModel(DbUtils.resultSetToTableModel(rs));
          //enable scroll bar
          JScrollPane scrollPane = new JScrollPane(book_list);
```

```
            f.add(scrollPane); //add scroll bar
            f.setSize(800, 400); //set dimensions of my books frame
            f.setVisible(true);
            f.setLocationRelativeTo(null);
        } catch (SQLException e1) {
            // TODO Auto-generated catch block
             JOptionPane.showMessageDialog(null, e1);
        }

    }
    }
    );



    f.add(my_book); //add my books
    f.add(view_but); // add view books
    f.setSize(300,100);//400 width and 500 height
    f.setLayout(null);//using no layout managers
    f.setVisible(true);//making the frame visible
    f.setLocationRelativeTo(null);
    }
public static void admin_menu() {


    JFrame f=new JFrame("Admin Functions"); //Give dialog box name as admin functions
    //f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); //


    JButton create_but=new JButton("Create/Reset");//creating instance of JButton to create or reset database
    create_but.setBounds(450,60,120,25);//x axis, y axis, width, height
    create_but.addActionListener(new ActionListener() { //Perform action
        public void actionPerformed(ActionEvent e){

            create(); //Call create function
            JOptionPane.showMessageDialog(null,"Database Created/Reset!"); //Open a dialog box and display
the message

        }
    });


    JButton view_but=new JButton("View Books");//creating instance of JButton to view books
    view_but.setBounds(20,20,120,25);//x axis, y axis, width, height
    view_but.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e){

            JFrame f = new JFrame("Books Available");
            //f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);


            Connection connection = connect(); //connect to database
            String sql="select * from BOOKS"; //select all books
        try {
            Statement stmt = connection.createStatement();
             stmt.executeUpdate("USE LIBRARY"); //use database
            stmt=connection.createStatement();
            ResultSet rs=stmt.executeQuery(sql);
            JTable book_list= new JTable(); //view data in table format
            book_list.setModel(DbUtils.resultSetToTableModel(rs));
            //mention scroll bar
            JScrollPane scrollPane = new JScrollPane(book_list);

            f.add(scrollPane); //add scrollpane
            f.setSize(800, 400); //set size for frame
            f.setVisible(true);
            f.setLocationRelativeTo(null);
        } catch (SQLException e1) {
            // TODO Auto-generated catch block
             JOptionPane.showMessageDialog(null, e1);
        }

}
}
);

JButton users_but=new JButton("View Users");//creating instance of JButton to view users
users_but.setBounds(150,20,120,25);//x axis, y axis, width, height
users_but.addActionListener(new ActionListener() { //Perform action on click button
    public void actionPerformed(ActionEvent e){

            JFrame f = new JFrame("Users List");
            //f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);


            Connection connection = connect();
            String sql="select * from users"; //retrieve all users
            try {
                Statement stmt = connection.createStatement();
                 stmt.executeUpdate("USE LIBRARY"); //use database
                stmt=connection.createStatement();
                ResultSet rs=stmt.executeQuery(sql);
                JTable book_list= new JTable();
                book_list.setModel(DbUtils.resultSetToTableModel(rs));
                //mention scroll bar
                JScrollPane scrollPane = new JScrollPane(book_list);

                f.add(scrollPane); //add scrollpane
                f.setSize(800, 400); //set size for frame
                f.setVisible(true);
                f.setLocationRelativeTo(null);
            } catch (SQLException e1) {
                // TODO Auto-generated catch block
                 JOptionPane.showMessageDialog(null, e1);
            }
    }
       }
    );

    JButton issued_but=new JButton("View Issued Books");//creating instance of JButton to view the issued books
    issued_but.setBounds(280,20,160,25);//x axis, y axis, width, height
    issued_but.addActionListener(new ActionListener() {
       public void actionPerformed(ActionEvent e){

             JFrame f = new JFrame("Users List");
             //f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);


             Connection connection = connect();
             String sql="select * from issued";
             try {
                Statement stmt = connection.createStatement();
                 stmt.executeUpdate("USE LIBRARY");
                stmt=connection.createStatement();
                ResultSet rs=stmt.executeQuery(sql);
                JTable book_list= new JTable();
                book_list.setModel(DbUtils.resultSetToTableModel(rs));

                JScrollPane scrollPane = new JScrollPane(book_list);

                f.add(scrollPane);
                f.setSize(800, 400);
                f.setVisible(true);
                f.setLocationRelativeTo(null);
             } catch (SQLException e1) {
                // TODO Auto-generated catch block
                 JOptionPane.showMessageDialog(null, e1);
             }

    }
       }
    );


    JButton add_user=new JButton("Add User"); //creating instance of JButton to add users
    add_user.setBounds(20,60,120,25); //set dimensions for button

    add_user.addActionListener(new ActionListener() {
       public void actionPerformed(ActionEvent e){

             JFrame g = new JFrame("Enter User Details"); //Frame to enter user details
             //g.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
             //Create label
             JLabel l1,l2;
             l1=new JLabel("Username");  //label 1 for username
             l1.setBounds(30,15, 100,30);
```

```
        l2=new JLabel("Password");  //label 2 for password
        l2.setBounds(30,50, 100,30);

        //set text field for username
        JTextField F_user = new JTextField();
        F_user.setBounds(110, 15, 200, 30);

        //set text field for password
        JPasswordField F_pass=new JPasswordField();
        F_pass.setBounds(110, 50, 200, 30);
        //set radio button for admin
        JRadioButton a1 = new JRadioButton("Admin");
        a1.setBounds(55, 80, 200,30);
        //set radio button for user
        JRadioButton a2 = new JRadioButton("User");
        a2.setBounds(130, 80, 200,30);
        //add radio buttons
        ButtonGroup bg=new ButtonGroup();
        bg.add(a1);bg.add(a2);


        JButton create_but=new JButton("Create");//creating instance of JButton for Create
        create_but.setBounds(130,130,80,25);//x axis, y axis, width, height
        create_but.addActionListener(new ActionListener() {

          public void actionPerformed(ActionEvent e){

          String username = F_user.getText();
          String password = F_pass.getText();
          Boolean admin = false;

          if(a1.isSelected()) {
            admin=true;
          }

          Connection connection = connect();

          try {
          Statement stmt = connection.createStatement();
           stmt.executeUpdate("USE LIBRARY");
           stmt.executeUpdate("INSERT INTO USERS(USERNAME,PASSWORD,ADMIN) VALUES
('"+username+"','"+password+"',"+admin+")");
           JOptionPane.showMessageDialog(null,"User added!");
           g.dispose();

          }

          catch (SQLException e1) {
            // TODO Auto-generated catch block
             JOptionPane.showMessageDialog(null, e1);
          }

          }
        });


          g.add(create_but);
          g.add(a2);
          g.add(a1);
          g.add(l1);
          g.add(l2);
          g.add(F_user);
          g.add(F_pass);
          g.setSize(350,200);//400 width and 500 height
          g.setLayout(null);//using no layout managers
          g.setVisible(true);//making the frame visible
          g.setLocationRelativeTo(null);


}
});


JButton add_book=new JButton("Add Book"); //creating instance of JButton for adding books
add_book.setBounds(150,60,120,25);

add_book.addActionListener(new ActionListener() {
   public void actionPerformed(ActionEvent e){
        //set frame wot enter book details
        JFrame g = new JFrame("Enter Book Details");
        //g.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        // set labels
        JLabel l1,l2,l3;
        l1=new JLabel("Book Name");  //lebel 1 for book name
        l1.setBounds(30,15, 100,30);


        l2=new JLabel("Genre");  //label 2 for genre
        l2.setBounds(30,53, 100,30);

        l3=new JLabel("Price");  //label 2 for price
        l3.setBounds(30,90, 100,30);

        //set text field for book name
        JTextField F_bname = new JTextField();
        F_bname.setBounds(110, 15, 200, 30);

        //set text field for genre
        JTextField F_genre=new JTextField();
        F_genre.setBounds(110, 53, 200, 30);
        //set text field for price
        JTextField F_price=new JTextField();
        F_price.setBounds(110, 90, 200, 30);


        JButton create_but=new JButton("Submit");//creating instance of JButton to submit details
        create_but.setBounds(130,130,80,25);//x axis, y axis, width, height
```

```
                create_but.addActionListener(new ActionListener() {

                    public void actionPerformed(ActionEvent e){
                    // assign the book name, genre, price
                    String bname = F_bname.getText();
                    String genre = F_genre.getText();
                    String price = F_price.getText();
                    //convert price of integer to int
                    int price_int = Integer.parseInt(price);

                    Connection connection = connect();

                    try {
                    Statement stmt = connection.createStatement();
                     stmt.executeUpdate("USE LIBRARY");
                     stmt.executeUpdate("INSERT INTO BOOKS(BNAME,GENRE,PRICE) VALUES
('"+bname+"','"+genre+"',"+price_int+")");
                     JOptionPane.showMessageDialog(null,"Book added!");
                     g.dispose();

                    }

                    catch (SQLException e1) {
                        // TODO Auto-generated catch block
                         JOptionPane.showMessageDialog(null, e1);
                    }

                    }

                });

                    g.add(l3);
                    g.add(create_but);
                    g.add(l1);
                    g.add(l2);
                    g.add(F_bname);
                    g.add(F_genre);
                    g.add(F_price);
                    g.setSize(350,200);//400 width and 500 height
                    g.setLayout(null);//using no layout managers
                    g.setVisible(true);//making the frame visible
                    g.setLocationRelativeTo(null);

    }
    });


    JButton issue_book=new JButton("Issue Book"); //creating instance of JButton to issue books
    issue_book.setBounds(450,20,120,25);

    issue_book.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e){
                //enter details
                JFrame g = new JFrame("Enter Details");
```

```
//g.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
//create labels
JLabel l1,l2,l3,l4;
l1=new JLabel("Book ID(BID)");  // Label 1 for Book ID
l1.setBounds(30,15, 100,30);


l2=new JLabel("User ID(UID)");  //Label 2 for user ID
l2.setBounds(30,53, 100,30);

l3=new JLabel("Period(days)");  //Label 3 for period
l3.setBounds(30,90, 100,30);

l4=new JLabel("Issued Date(DD-MM-YYYY)");  //Label 4 for issue date
l4.setBounds(30,127, 150,30);

JTextField F_bid = new JTextField();
F_bid.setBounds(110, 15, 200, 30);


JTextField F_uid=new JTextField();
F_uid.setBounds(110, 53, 200, 30);

JTextField F_period=new JTextField();
F_period.setBounds(110, 90, 200, 30);

JTextField F_issue=new JTextField();
F_issue.setBounds(180, 130, 130, 30);


JButton create_but=new JButton("Submit");//creating instance of JButton
create_but.setBounds(130,170,80,25);//x axis, y axis, width, height
create_but.addActionListener(new ActionListener() {

   public void actionPerformed(ActionEvent e){

   String uid = F_uid.getText();
   String bid = F_bid.getText();
   String period = F_period.getText();
   String issued_date = F_issue.getText();

   int period_int = Integer.parseInt(period);

   Connection connection = connect();

   try {
   Statement stmt = connection.createStatement();
    stmt.executeUpdate("USE LIBRARY");
    stmt.executeUpdate("INSERT INTO ISSUED(UID,BID,ISSUED_DATE,PERIOD) VALUES
('"+uid+"','"+bid+"','"+issued_date+"',"+period_int+")");
    JOptionPane.showMessageDialog(null,"Book Issued!");
    g.dispose();

   }
```

```
                catch (SQLException e1) {
                    // TODO Auto-generated catch block
                     JOptionPane.showMessageDialog(null, e1);
                }

                }

            });


                g.add(l3);
                g.add(l4);
                g.add(create_but);
                g.add(l1);
                g.add(l2);
                g.add(F_uid);
                g.add(F_bid);
                g.add(F_period);
                g.add(F_issue);
                g.setSize(350,250);//400 width and 500 height
                g.setLayout(null);//using no layout managers
                g.setVisible(true);//making the frame visible
                g.setLocationRelativeTo(null);


}
});


JButton return_book=new JButton("Return Book"); //creating instance of JButton to return books
return_book.setBounds(280,60,160,25);

return_book.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e){

            JFrame g = new JFrame("Enter Details");
            //g.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
            //set labels
            JLabel l1,l2,l3,l4;
            l1=new JLabel("Issue ID(IID)");  //Label 1 for Issue ID
            l1.setBounds(30,15, 100,30);


            l4=new JLabel("Return Date(DD-MM-YYYY)");
            l4.setBounds(30,50, 150,30);

            JTextField F_iid = new JTextField();
            F_iid.setBounds(110, 15, 200, 30);


            JTextField F_return=new JTextField();
            F_return.setBounds(180, 50, 130, 30);
```

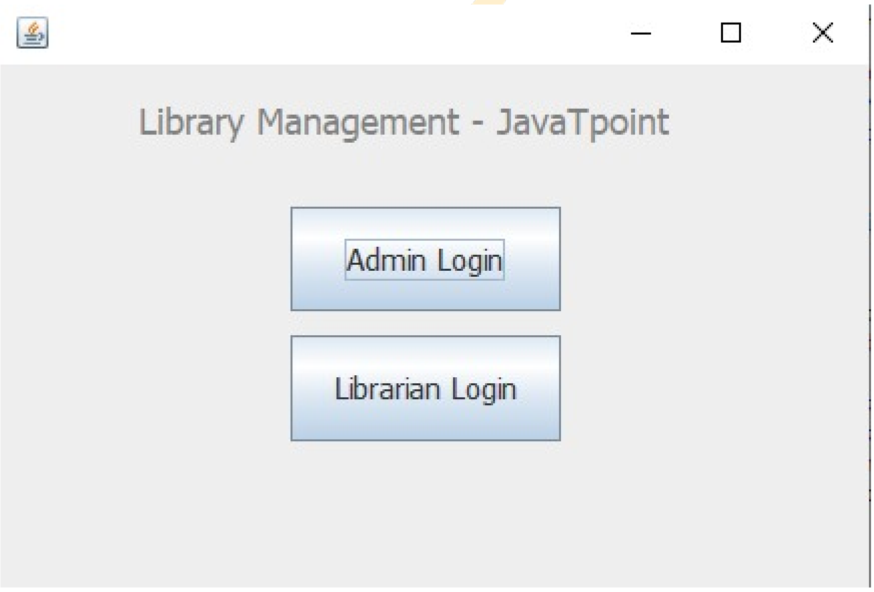
```
        JButton create_but=new JButton("Return");//creating instance of JButton to mention return date and
calculcate fine
        create_but.setBounds(130,170,80,25);//x axis, y axis, width, height
        create_but.addActionListener(new ActionListener() {

          public void actionPerformed(ActionEvent e){

          String iid = F_iid.getText();
          String return_date = F_return.getText();

          Connection connection = connect();

          try {
          Statement stmt = connection.createStatement();
           stmt.executeUpdate("USE LIBRARY");
           //Intialize date1 with NULL value
           String date1=null;
           String date2=return_date; //Intialize date2 with return date

           //select issue date
           ResultSet rs = stmt.executeQuery("SELECT ISSUED_DATE FROM ISSUED WHERE
IID="+iid);
           while (rs.next()) {
             date1 = rs.getString(1);

            }

           try {
               Date date_1=new SimpleDateFormat("dd-MM-yyyy").parse(date1);
               Date date_2=new SimpleDateFormat("dd-MM-yyyy").parse(date2);
               //subtract the dates and store in diff
               long diff = date_2.getTime() - date_1.getTime();
               //Convert diff from milliseconds to days
               ex.days=(int)(TimeUnit.DAYS.convert(diff, TimeUnit.MILLISECONDS));


             } catch (ParseException e1) {
               // TODO Auto-generated catch block
               e1.printStackTrace();
             }


           //update return date
           stmt.executeUpdate("UPDATE ISSUED SET RETURN_DATE='"+return_date+"' WHERE
IID="+iid);
           g.dispose();


           Connection connection1 = connect();
           Statement stmt1 = connection1.createStatement();
           stmt1.executeUpdate("USE LIBRARY");
          ResultSet rs1 = stmt1.executeQuery("SELECT PERIOD FROM ISSUED WHERE IID="+iid);
//set period
```

```
                String diff=null;
                while (rs1.next()) {
                    diff = rs1.getString(1);

                  }
                int diff_int = Integer.parseInt(diff);
                if(ex.days&amp;amp;amp;amp;amp;amp;amp;amp;amp;amp;gt;diff_int) { //If number of days are
more than the period then calculcate fine

                   //System.out.println(ex.days);
                   int fine = (ex.days-diff_int)*10; //fine for every day after the period is Rs 10.
                   //update fine in the system
                   stmt1.executeUpdate("UPDATE ISSUED SET FINE="+fine+" WHERE IID="+iid);
                   String fine_str = ("Fine: Rs. "+fine);
                   JOptionPane.showMessageDialog(null,fine_str);

                }

                 JOptionPane.showMessageDialog(null,"Book Returned!");

                }


                catch (SQLException e1) {
                   // TODO Auto-generated catch block
                    JOptionPane.showMessageDialog(null, e1);
                }

                }

             });
                g.add(l4);
                g.add(create_but);
                g.add(l1);
                g.add(F_iid);
                g.add(F_return);
                g.setSize(350,250);//400 width and 500 height
                g.setLayout(null);//using no layout managers
                g.setVisible(true);//making the frame visible
                g.setLocationRelativeTo(null);
   }
   });

   f.add(create_but);
   f.add(return_book);
   f.add(issue_book);
   f.add(add_book);
   f.add(issued_but);
   f.add(users_but);
   f.add(view_but);
   f.add(add_user);
   f.setSize(600,200);//400 width and 500 height
   f.setLayout(null);//using no layout managers
   f.setVisible(true);//making the frame visible
```

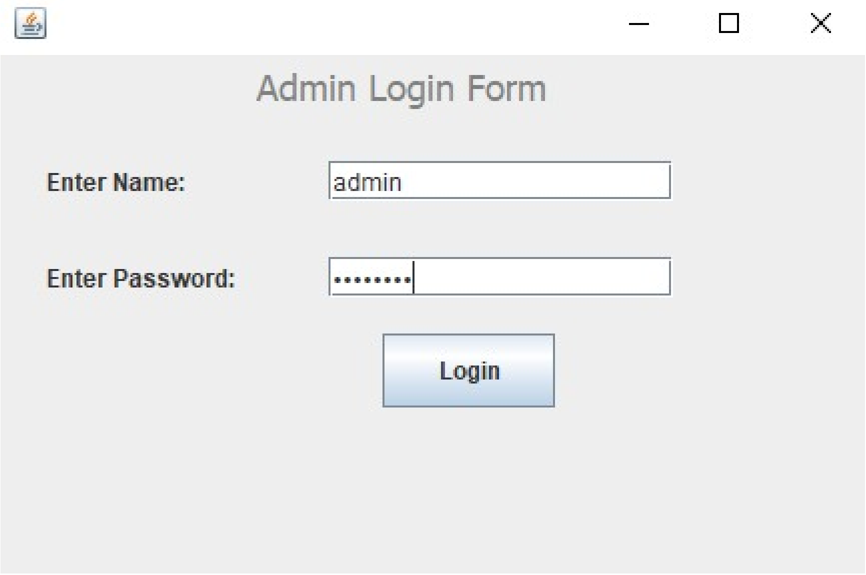
```
    f.setLocationRelativeTo(null);

    }
}
```

Output:

INITIAL FRAME: MAIN LOGIN PAGE



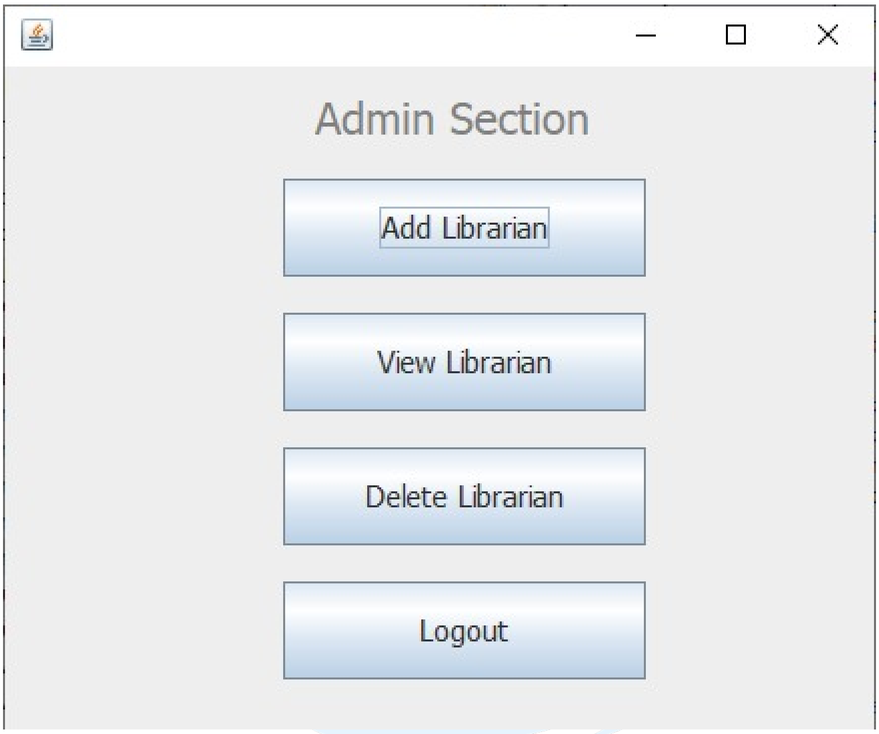
PART 1 – ADMIN lOGIN



Enter Name: admin

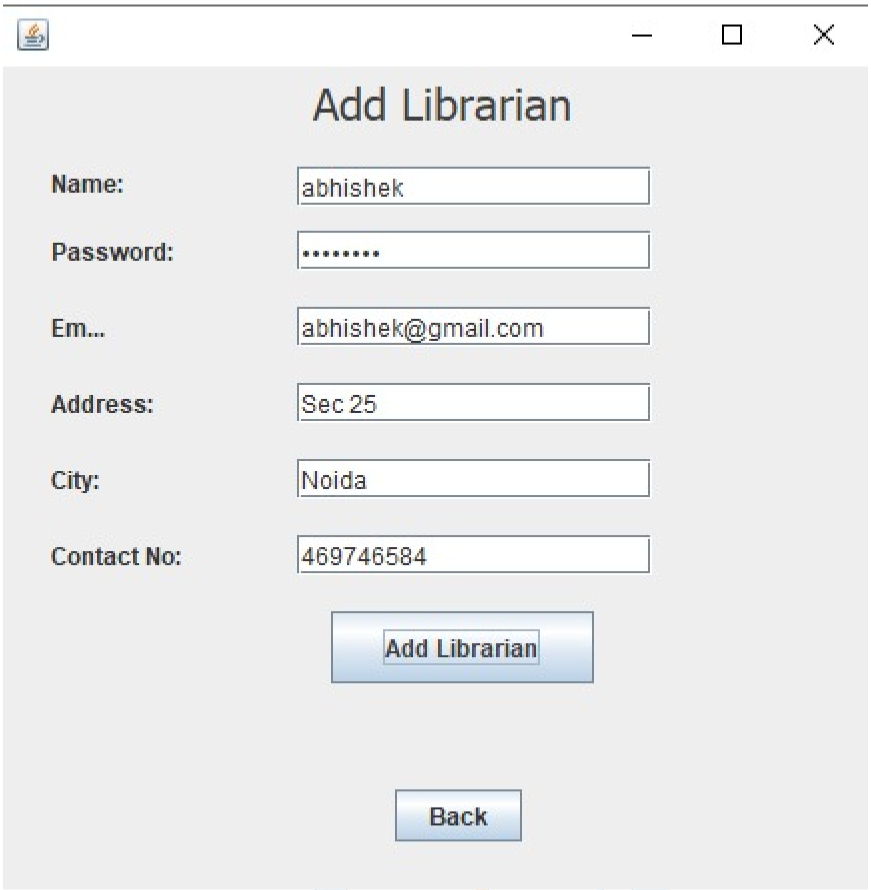
Enter Password: ••••••••

ADMIN SECTION



ADMIN FEATURES

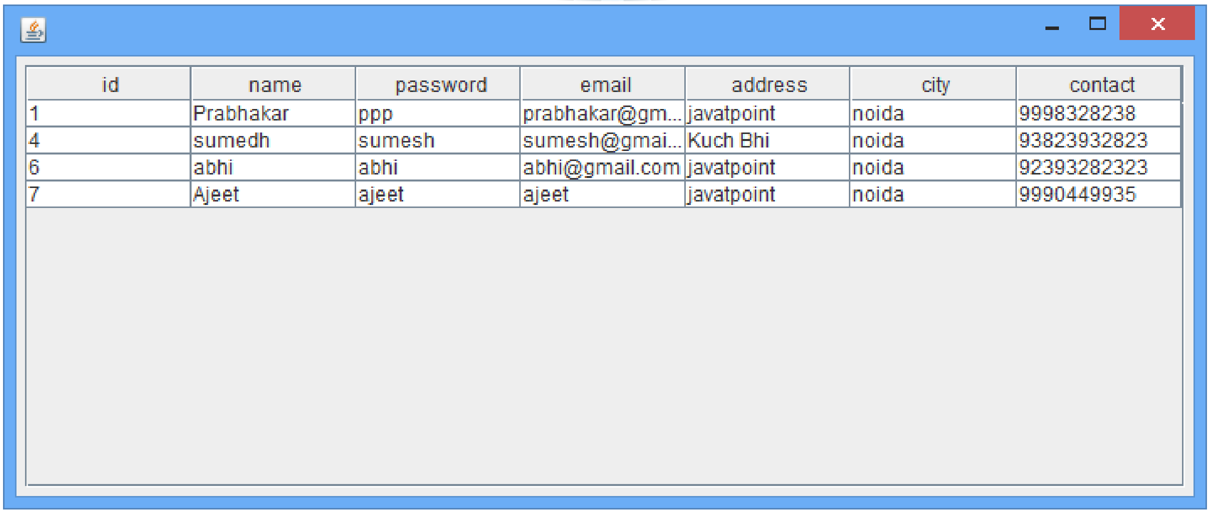
1. ADD LIBRARIAN



Add Librarian

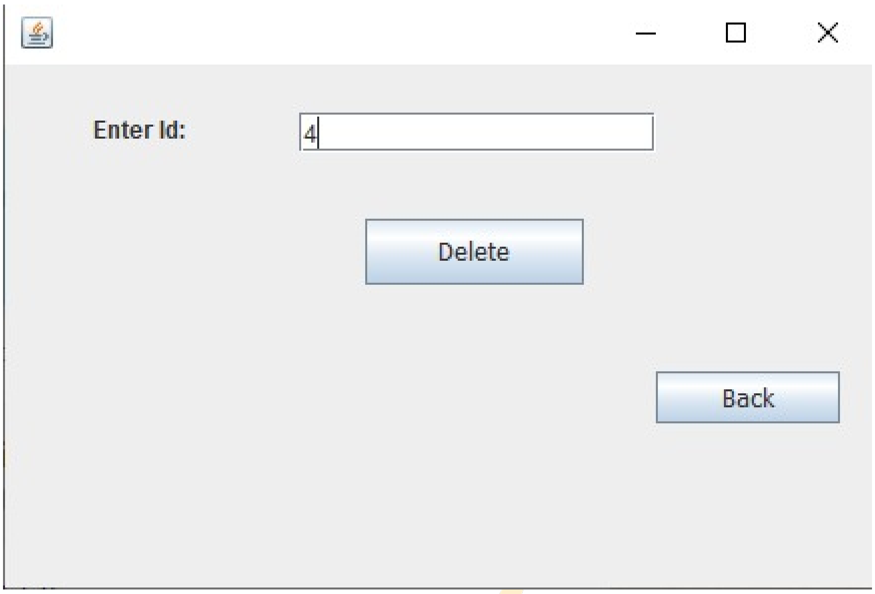
Name:	abhishek
Password:	•••••••••
Em...	abhishek@gmail.com
Address:	Sec 25
City:	Noida
Contact No:	469746584

2. VIEW LIBRARIAN

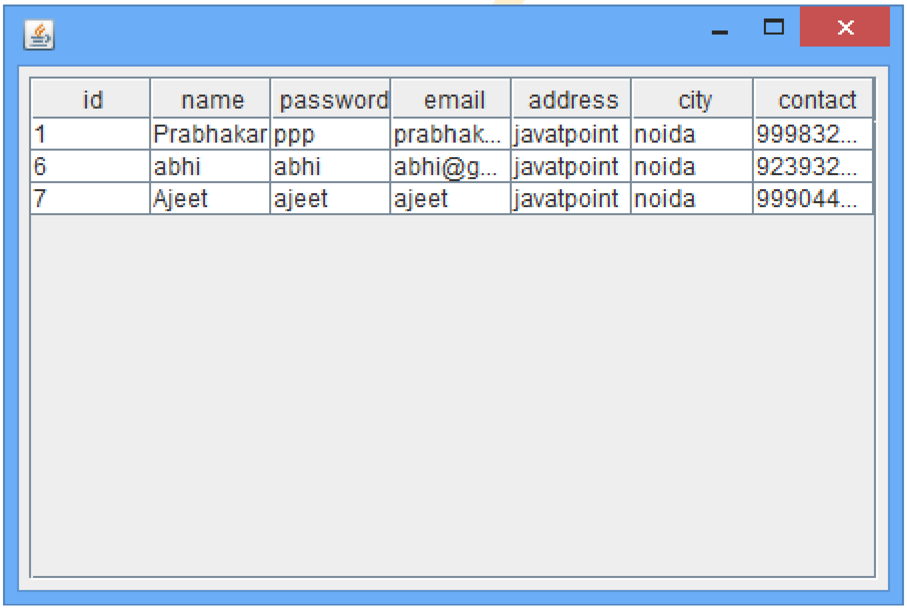


id	name	password	email	address	city	contact
1	Prabhakar	ppp	prabhakar@gm...	javatpoint	noida	9998328238
4	sumedh	sumesh	sumesh@gmai...	Kuch Bhi	noida	93823932823
6	abhi	abhi	abhi@gmail.com	javatpoint	noida	92393282323
7	Ajeet	ajeet	ajeet	javatpoint	noida	9990449935

3. DELETE LIBRARIAN

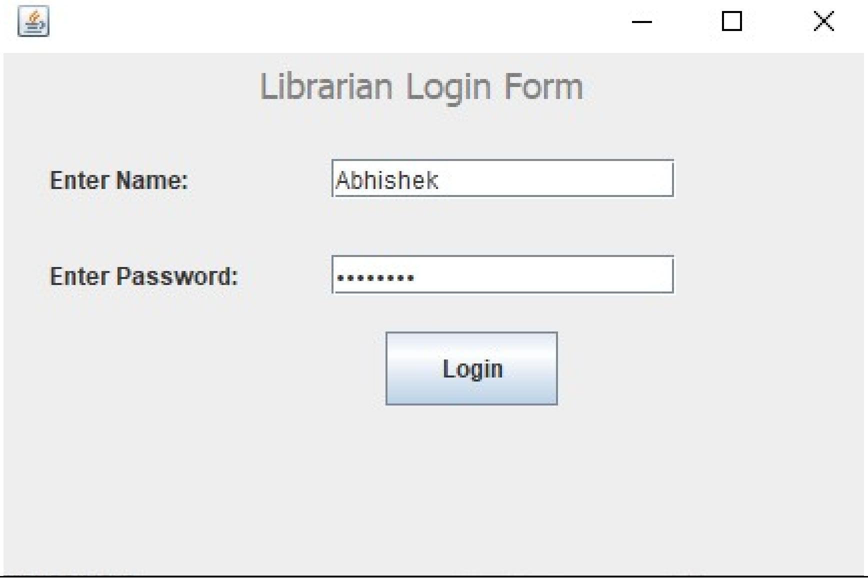


Enter Id: 4



id	name	password	email	address	city	contact
1	Prabhakar	ppp	prabhak...	javatpoint	noida	999832...
6	abhi	abhi	abhi@g...	javatpoint	noida	923932...
7	Ajeet	ajeet	ajeet	javatpoint	noida	999044...

PART 2 – LIBRARIAN LOGIN

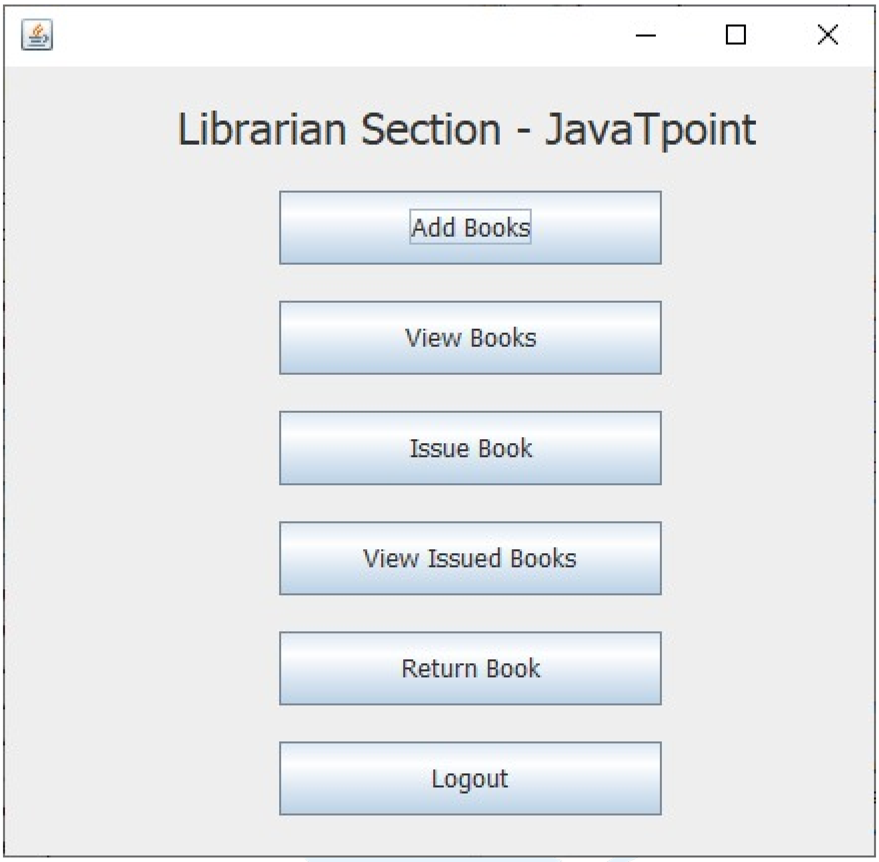


Librarian Login Form

Enter Name: Abhishek

Enter Password: ••••••••

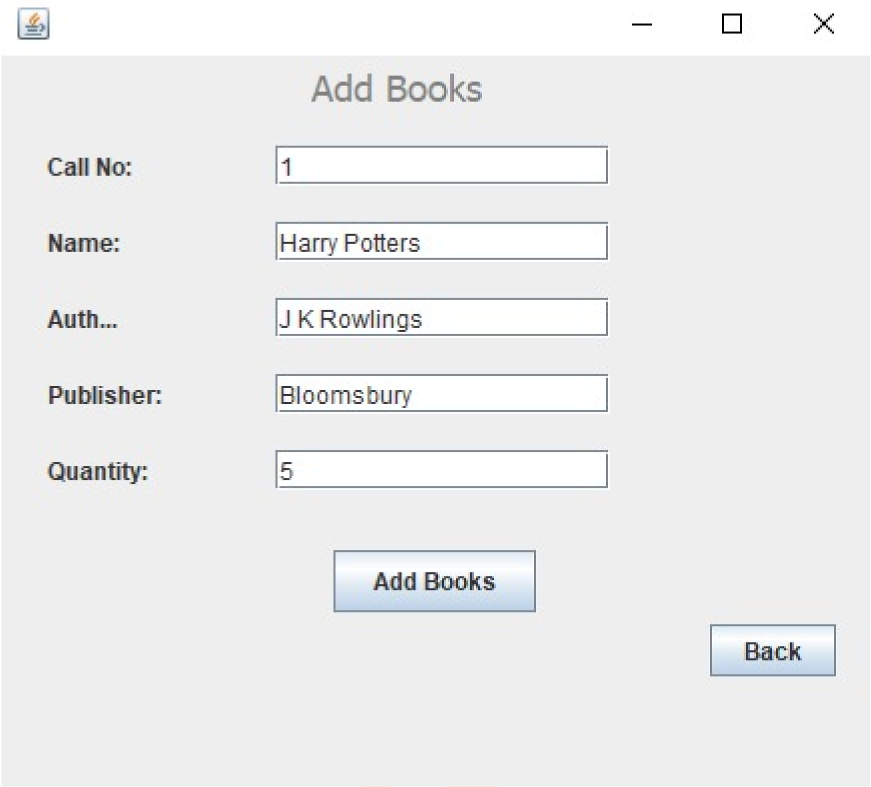
LIBRARIAN SECTION



Librarian Section - JavaTpoint

LIBRARIAN FEATURES

1. ADD BOOKS



Add Books

Call No: 1

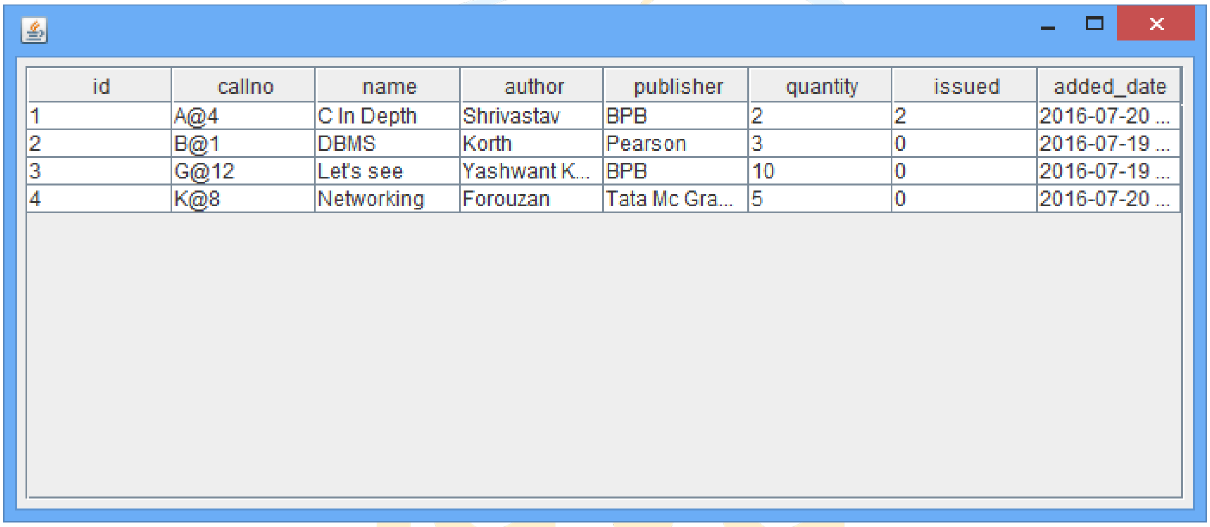
Name: Harry Potters

Auth... J K Rowlings

Publisher: Bloomsbury

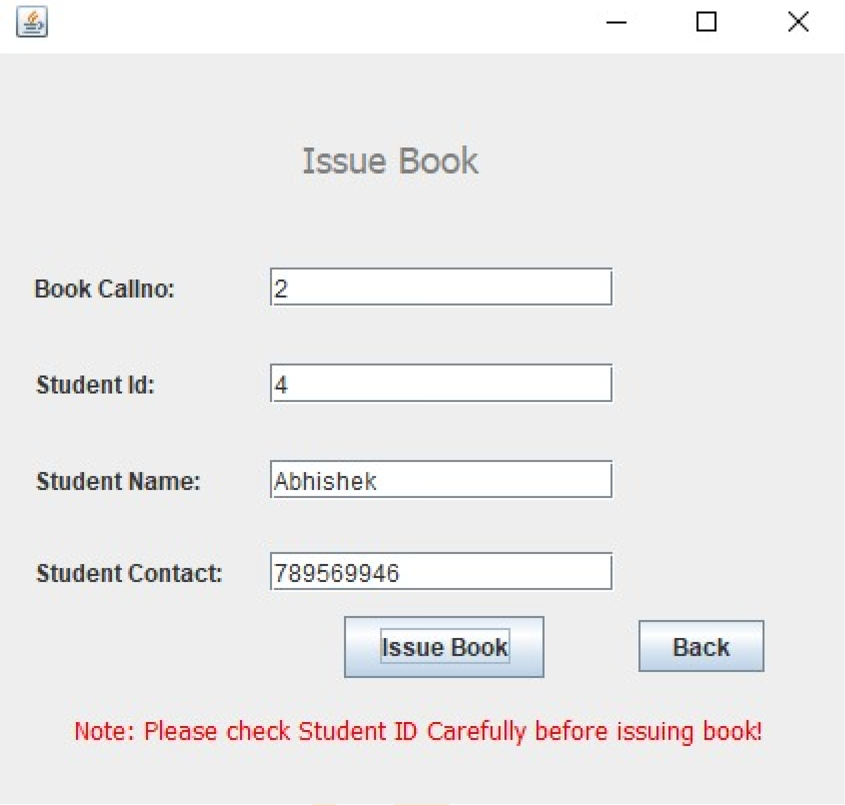
Quantity: 5

2. VIEW BOOKS



id	callno	name	author	publisher	quantity	issued	added_date
1	A@4	C In Depth	Shrivastav	BPB	2	2	2016-07-20 ...
2	B@1	DBMS	Korth	Pearson	3	0	2016-07-19 ...
3	G@12	Let's see	Yashwant K...	BPB	10	0	2016-07-19 ...
4	K@8	Networking	Forouzan	Tata Mc Gra...	5	0	2016-07-20 ...

3. ISSUE BOOKS



Issue Book

Book Callno: 2

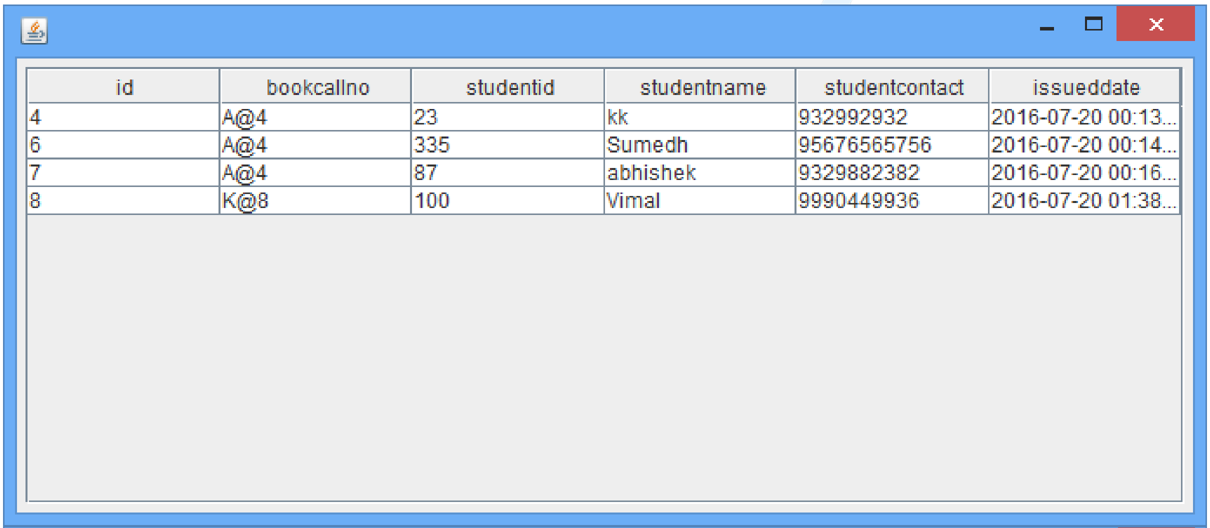
Student Id: 4

Student Name: Abhishek

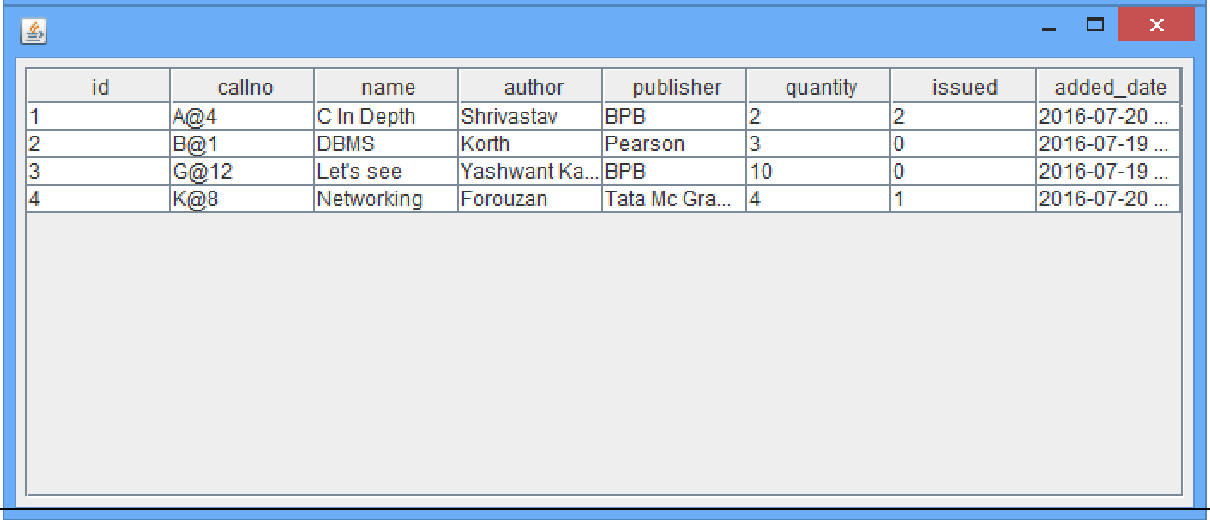
Student Contact: 789569946

Note: Please check Student ID Carefully before issuing book!

4. VIEW ISSUED BOOKS

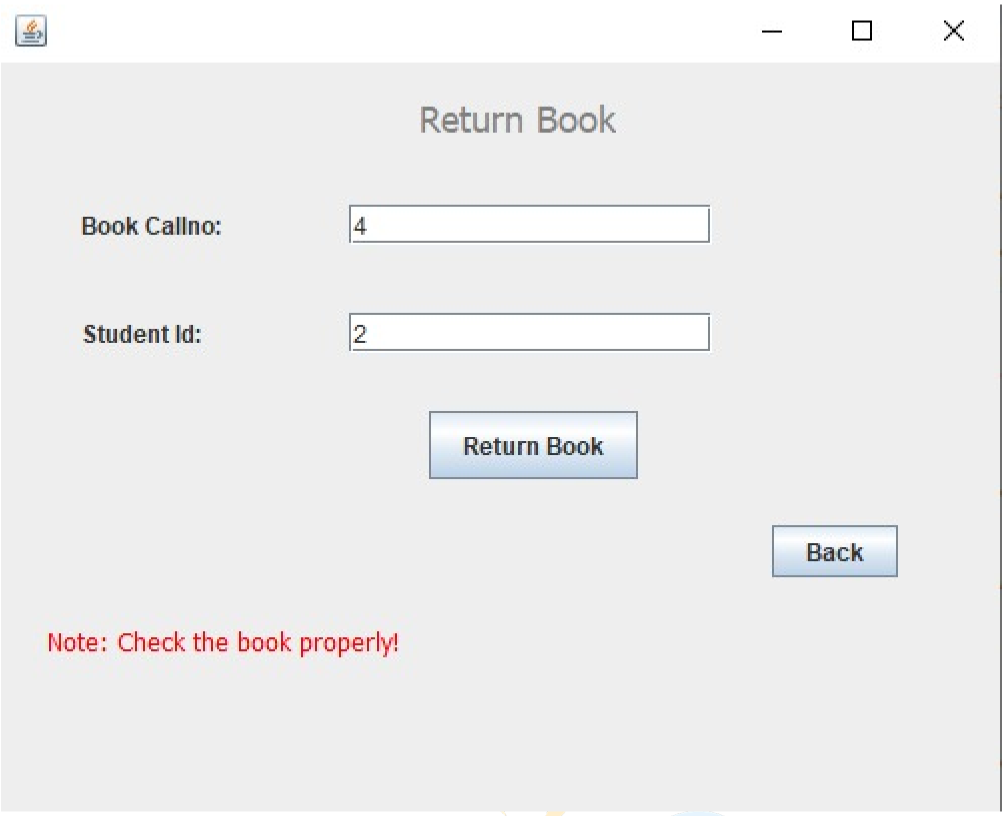


id	bookcallno	studentid	studentname	studentcontact	issueddate
4	A@4	23	kk	932992932	2016-07-20 00:13...
6	A@4	335	Sumedh	95676565756	2016-07-20 00:14...
7	A@4	87	abhishek	9329882382	2016-07-20 00:16...
8	K@8	100	Vimal	9990449936	2016-07-20 01:38...



id	callno	name	author	publisher	quantity	issued	added_date
1	A@4	C In Depth	Shrivastav	BPB	2	2	2016-07-20 ...
2	B@1	DBMS	Korth	Pearson	3	0	2016-07-19 ...
3	G@12	Let's see	Yashwant Ka...	BPB	10	0	2016-07-19 ...
4	K@8	Networking	Forouzan	Tata Mc Gra...	4	1	2016-07-20 ...

5. RETURN BOOKS



Return Book

Book Callno: 4

Student Id: 2

Note: Check the book properly!