SZABIST

Object Oriented Programming

PROJECT REPORT

Project = Library Management System

GROUP MEMBERS

Ilyaan Umatia – (2212288)

Salman Hamzo – (2212303)

Ovais Salam – (2212297)

ACKNOWLEDGMENT

We are really grateful because we managed to complete our project within the time given by our teachers [Syed Muhammad Hassan & Bakhtawar Abbasi]. This project cannot be completed without the effort and cooperation of our group members, Group members [Ilyaan Umatia, Salman Hamzo & Ovais Salam].

We also sincerely thank our teachers [Syed Muhammad Hassan & Bakhtawar Abbasi] for their guidance and encouragement in finishing this project and also for teaching us in this course.

Object Oriented Programming

Library Management System

PROJECT DESCRIPTION

Our Library Management System is an automated software solution designed to efficiently manage library operations, resources, and services. This project aims to develop a user-friendly and scalable application that caters specifically to the needs of a library.

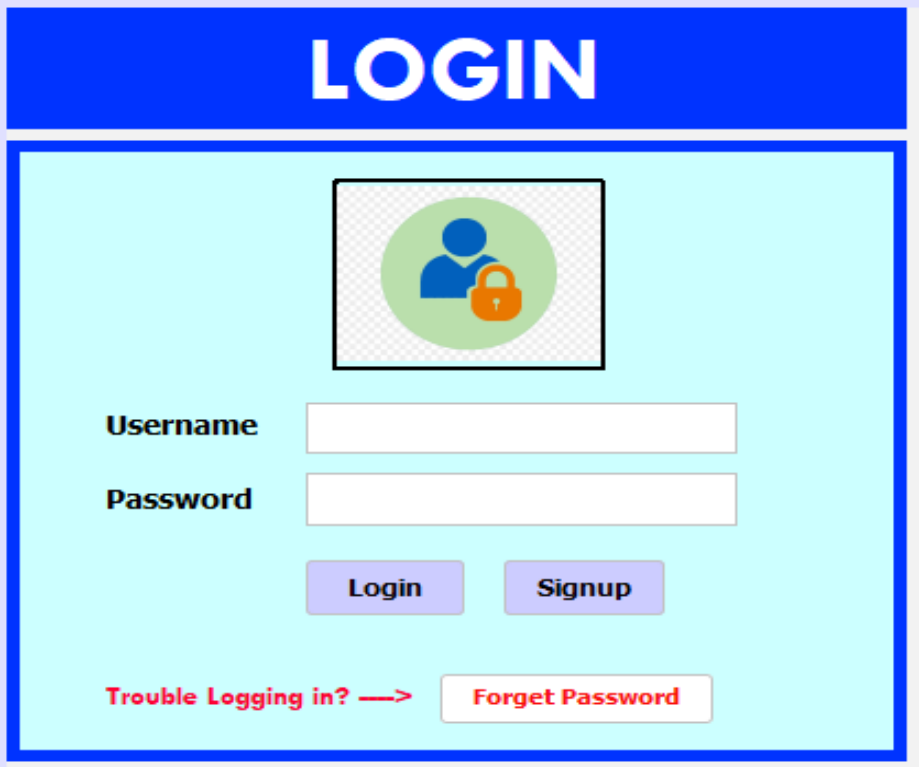
It aims to improve library operations, access to resources, and enhance user satisfaction. By automating various processes, it will save time, reduce manual effort, and provide a good experience for both librarians and students.

Functionalities & Features:

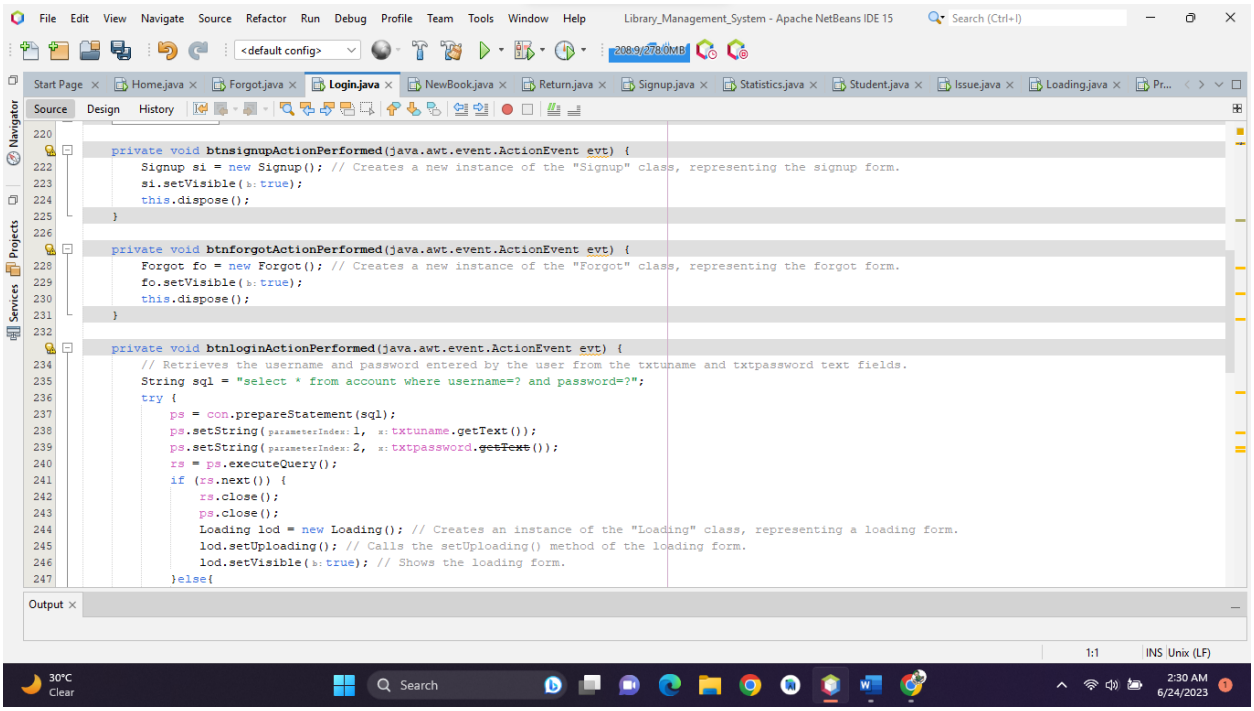
- **Login System:** Admin account for the librarian to access the records and operations.
- **Add Book:** Easily add new books to the library's database.
- **Records:** Maintain comprehensive records of books, students, and transactions.
- **Add Student:** Effortlessly add new students to the library system.
- **Issue Book:** Facilitate the process of borrowing books from the library.
- **Return Book:** Organized return process for borrowed books.
- **Print Library Card:** Generate library cards for students to access library resources.

SOURCE CODE AND FORMS

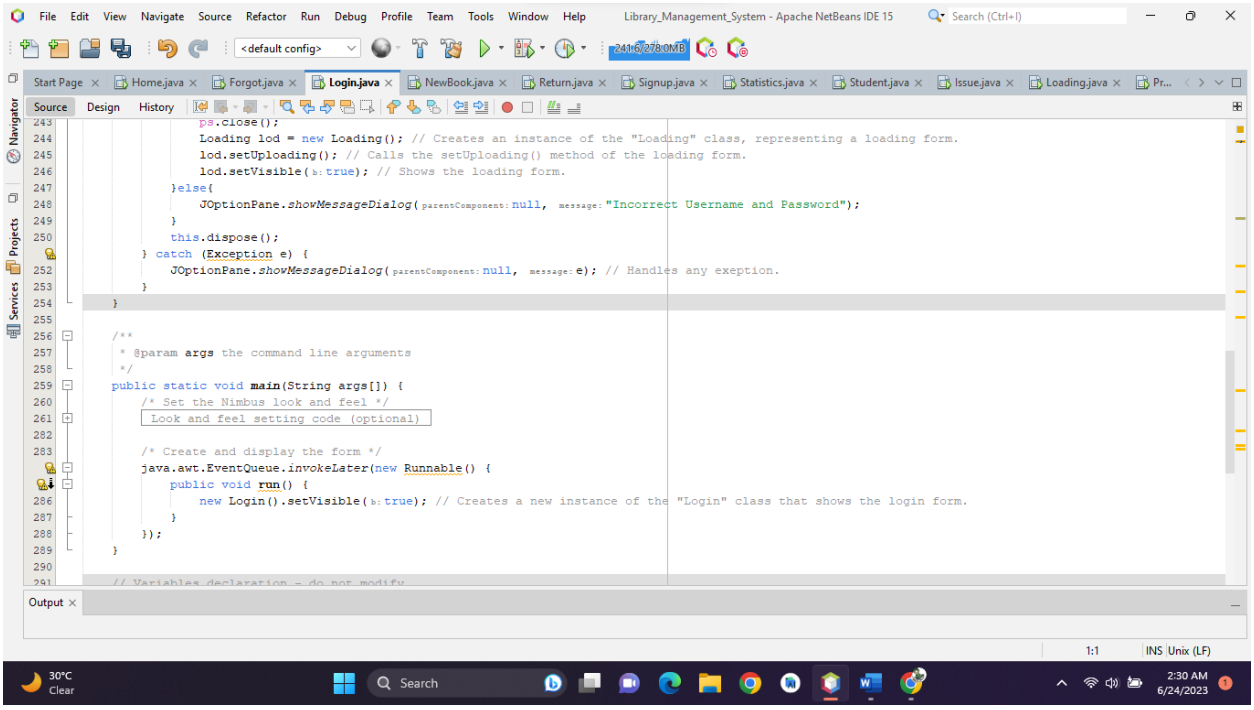
Login Form:




```
package Library_Management;

import DatabaseConnector.DatabaseHelper;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import javax.swing.JOptionPane;


public class Login extends javax.swing.JFrame {
    Connection con = null;
    ResultSet rs = null; // Declaration and initialization of variables for database operations.
    PreparedStatement ps = null;
    /**
     * Creates new form Login
     */
    public Login() {
        initComponents();
        con = DatabaseHelper.getConnection();

    }

    /**
     * This method is called from within the constructor to initialize the form.
     * WARNING: Do NOT modify this code. The content of this method is always
     * regenerated by the Form Editor.
     */
```



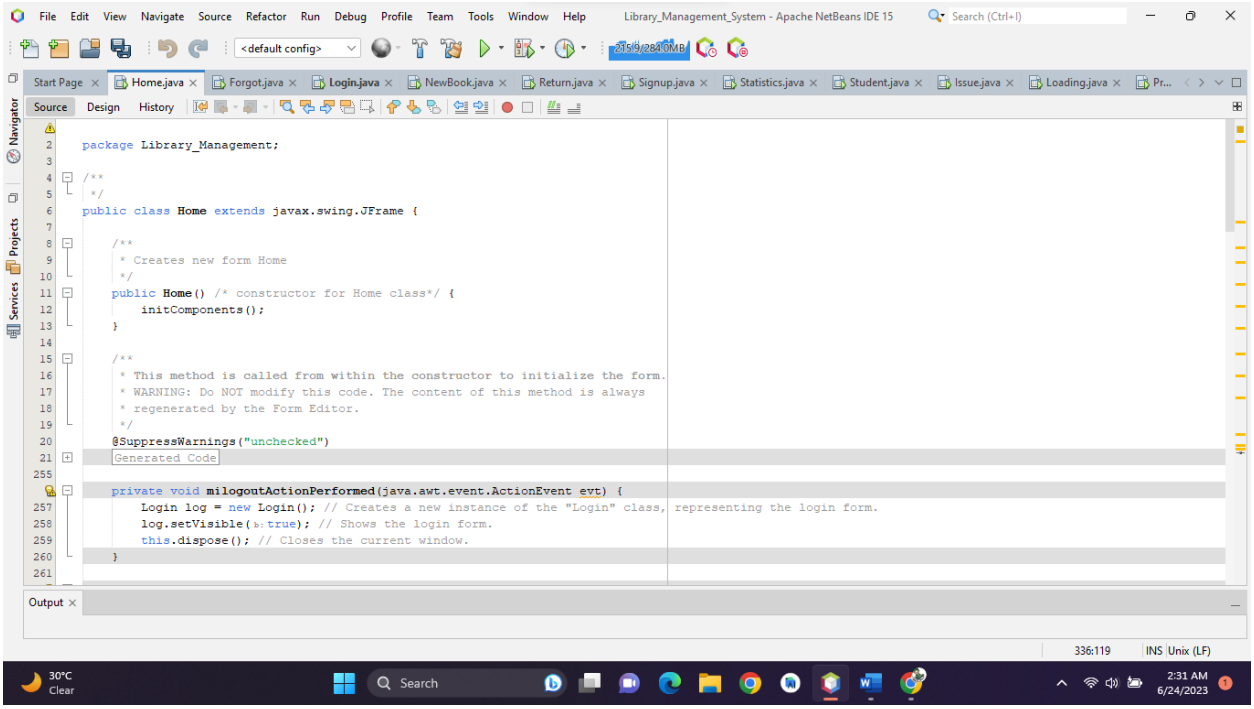
```
220
221
222     private void btnsignupActionPerformed(java.awt.event.ActionEvent evt) {
223         Signup si = new Signup(); // Creates a new instance of the "Signup" class, representing the signup form.
224         si.setVisible(b: true);
225         this.dispose();
226     }
227
228     private void btnforgotActionPerformed(java.awt.event.ActionEvent evt) {
229         Forgot fo = new Forgot(); // Creates a new instance of the "Forgot" class, representing the forgot form.
230         fo.setVisible(b: true);
231         this.dispose();
232     }
233
234     private void btnloginActionPerformed(java.awt.event.ActionEvent evt) {
235         // Retrieves the username and password entered by the user from the txtuname and txtpassword text fields.
236         String sql = "select * from account where username=? and password=?";
237         try {
238             ps = con.prepareStatement(sql);
239             ps.setString(parameterIndex: 1, x: txtuname.getText());
240             ps.setString(parameterIndex: 2, x: txtpassword.getText());
241             rs = ps.executeQuery();
242             if (rs.next()) {
243                 rs.close();
244                 ps.close();
245                 Loading lod = new Loading(); // Creates an instance of the "Loading" class, representing a loading form.
246                 lod.setUploading(); // Calls the setUploading() method of the loading form.
247                 lod.setVisible(b: true); // Shows the loading form.
                }else{
```

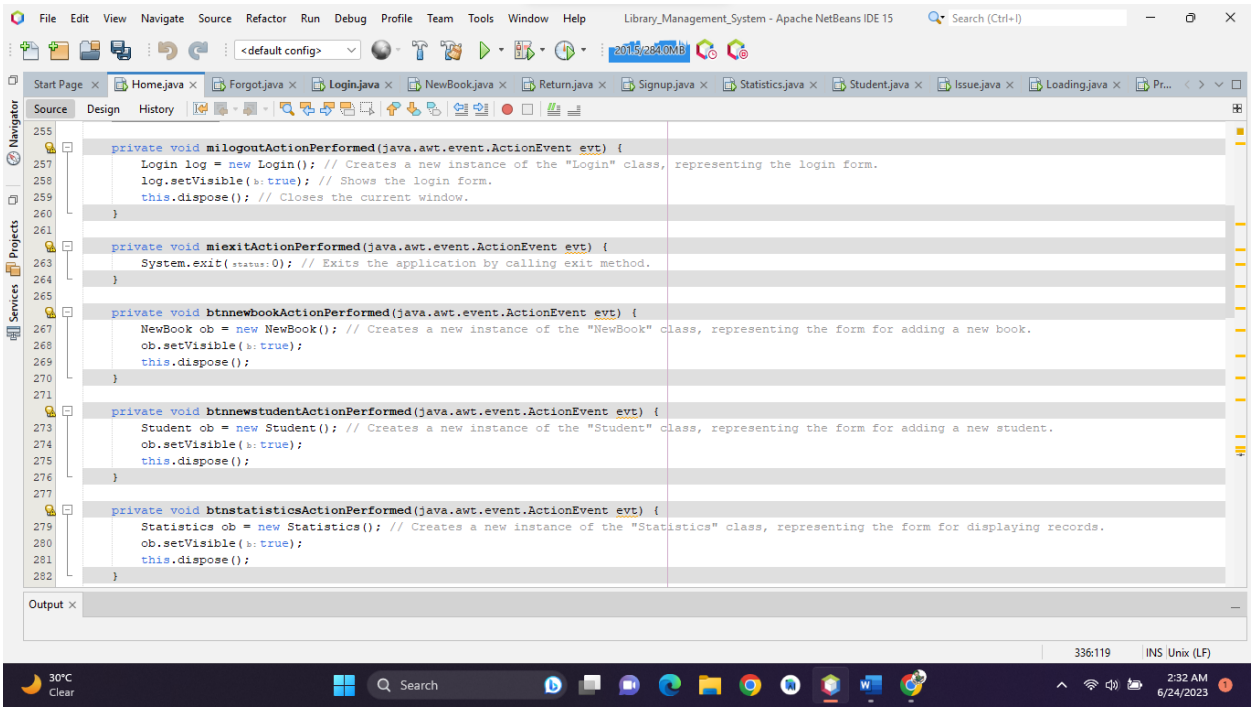


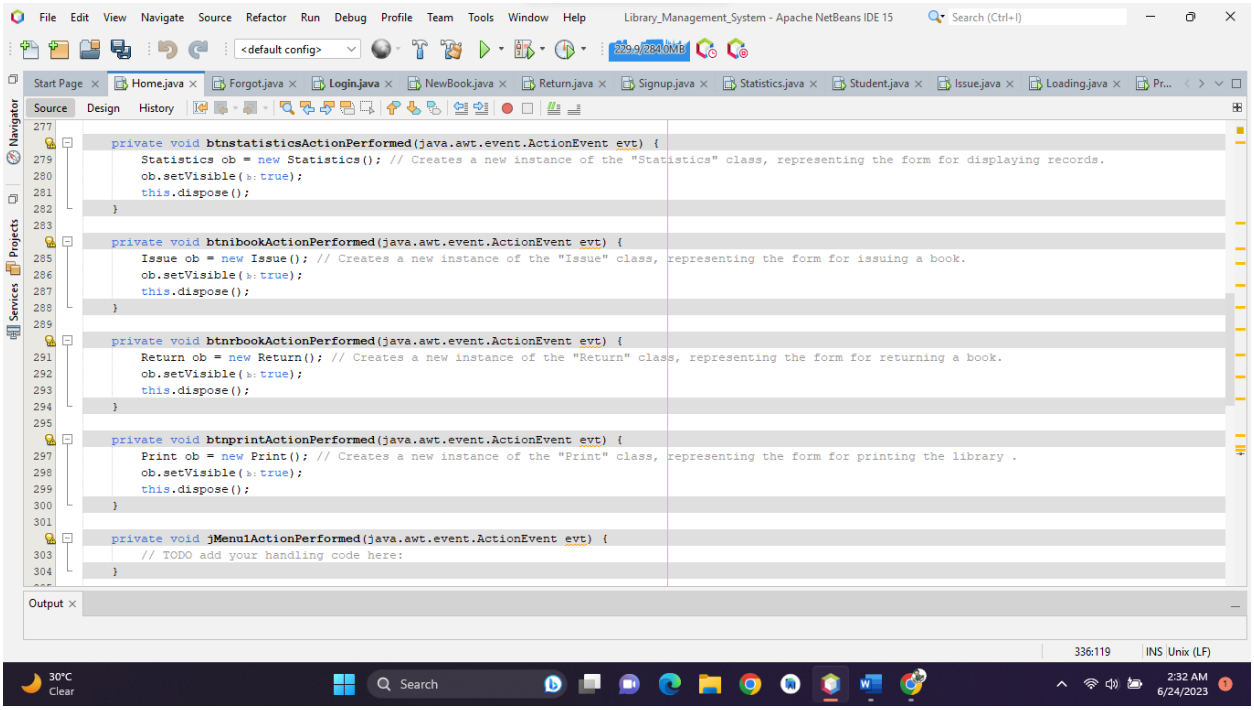
```
243                 ps.close();
244                 Loading lod = new Loading(); // Creates an instance of the "Loading" class, representing a loading form.
245                 lod.setUploading(); // Calls the setUploading() method of the loading form.
246                 lod.setVisible(b: true); // Shows the loading form.
247             }else{
248                 JOptionPane.showMessageDialog(parentComponent: null, message: "Incorrect Username and Password");
249             }
250             this.dispose();
252         } catch (Exception e) {
253             JOptionPane.showMessageDialog(parentComponent: null, message: e); // Handles any exeption.
254         }
255     }
256
257     /**
258      * @param args the command line arguments
259      */
260     public static void main(String args[]) {
261         /* Set the Nimbus look and feel */
262         Look and feel setting code (optional)
282
283         /* Create and display the form */
284         java.awt.EventQueue.invokeLater(new Runnable() {
285             public void run() {
286                 new Login().setVisible(b: true); // Creates a new instance of the "Login" class that shows the login form.
287             }
288         });
289     }
290
291     // Variables declaration - do not modify
```

Home Form:


Main Menu
LIBRARY MANAGEMENT SYSTEM
SZABIST
Add Book
Records
Add Student
Issue Book
Return Book
Print Card

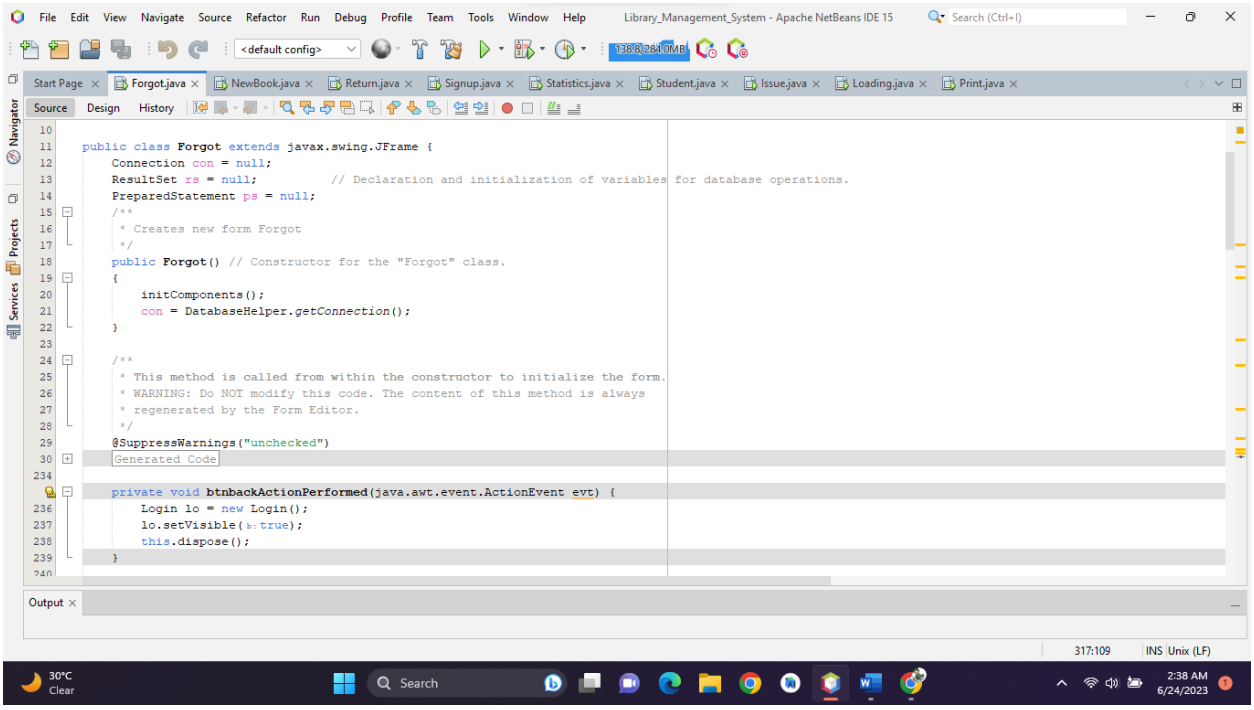

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Library_Management_System - Apache NetBeans IDE 15
Start Page Home.java Forgot.java Login.java NewBook.java Return.java Signup.java Statistics.java Student.java Issue.java Loading.java
Source Design History
package Library_Management;
public class Home extends javax.swing.JFrame {
Creates new form Home
public Home() /* constructor for Home class*/ {
initComponents();
This method is called from within the constructor to initialize the form.
WARNING: Do NOT modify this code. The content of this method is always
regenerated by the Form Editor.
@SuppressWarnings("unchecked")
Generated Code
private void milogoutActionPerformed(java.awt.event.ActionEvent evt) {
Login log = new Login(); // Creates a new instance of the "Login" class, representing the login form.
log.setVisible(b:true); // Shows the login form.
this.dispose(); // Closes the current window.
Output
336:119 INS Unix (LF)
30°C Clear
Search
2:31 AM 6/24/2023

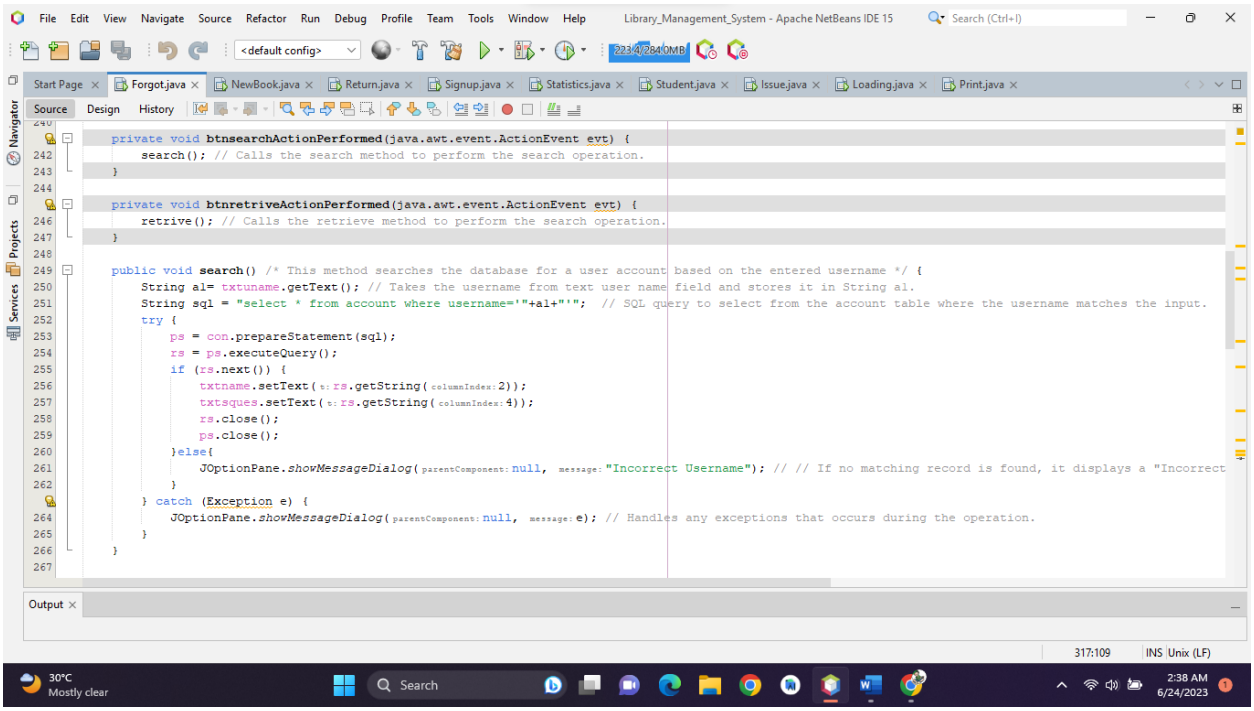

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Library_Management_System - Apache NetBeans IDE 15
Search (Ctrl+I)
<default config>
Start Page
Home.java
Forgot.java
Login.java
NewBook.java
Return.java
Signup.java
Statistics.java
Student.java
Issue.java
Loading.java
Source
Design
History
255 private void milogoutActionPerformed(java.awt.event.ActionEvent evt) {
257 Login log = new Login(); // Creates a new instance of the "Login" class, representing the login form.
258 log.setVisible(b: true); // Shows the login form.
259 this.dispose(); // Closes the current window.
260 }
261
263 private void miexitActionPerformed(java.awt.event.ActionEvent evt) {
264 System.exit(status: 0); // Exits the application by calling exit method.
265 }
267 private void btnnewbookActionPerformed(java.awt.event.ActionEvent evt) {
268 NewBook ob = new NewBook(); // Creates a new instance of the "NewBook" class, representing the form for adding a new book.
269 ob.setVisible(b: true);
270 this.dispose();
271 }
273 private void btnnewstudentActionPerformed(java.awt.event.ActionEvent evt) {
274 Student ob = new Student(); // Creates a new instance of the "Student" class, representing the form for adding a new student.
275 ob.setVisible(b: true);
276 this.dispose();
277 }
279 private void btnstatisticsActionPerformed(java.awt.event.ActionEvent evt) {
280 Statistics ob = new Statistics(); // Creates a new instance of the "Statistics" class, representing the form for displaying records.
281 ob.setVisible(b: true);
282 this.dispose();
283 }
Output
336:119
INS Unix (LF)
30°C
Clear
Search
2:32 AM
6/24/2023

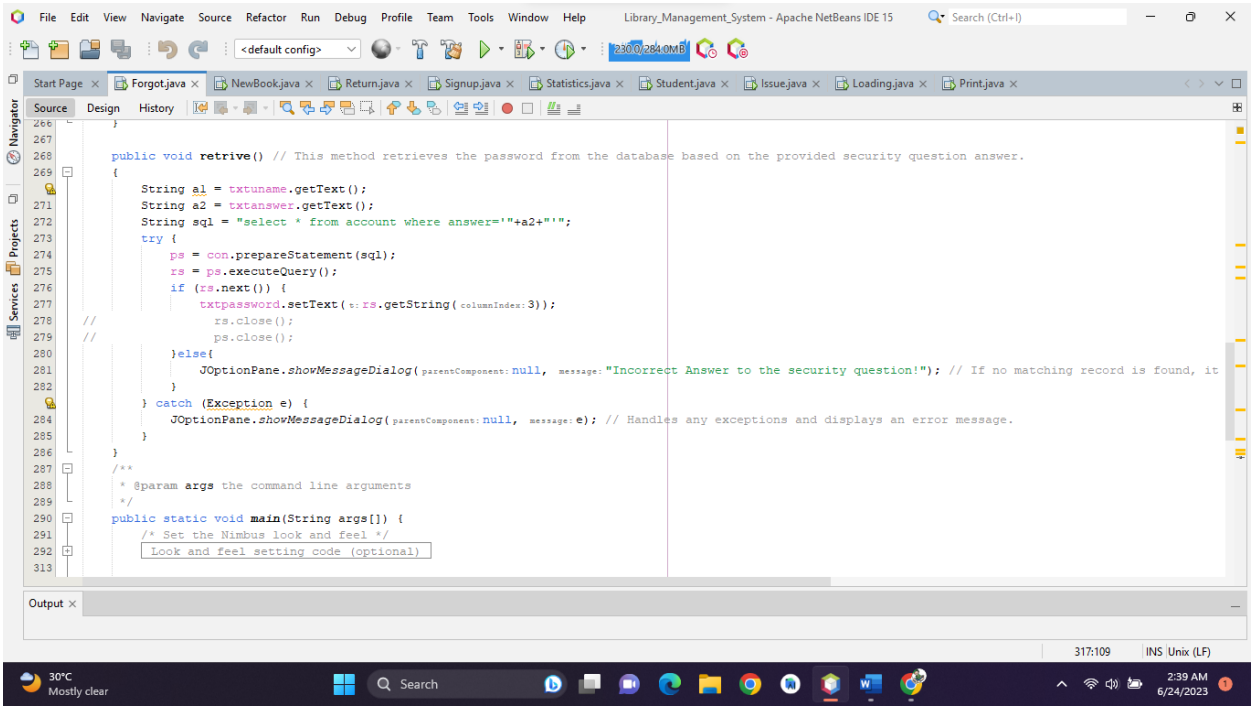

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Library_Management_System - Apache NetBeans IDE 15
Search (Ctrl+I)
<default config>
Start Page
Home.java
Forgot.java
Login.java
NewBook.java
Return.java
Signup.java
Statistics.java
Student.java
Issue.java
Loading.java
Source
Design
History
277
279 private void btnstatisticsActionPerformed(java.awt.event.ActionEvent evt) {
280 Statistics ob = new Statistics(); // Creates a new instance of the "Statistics" class, representing the form for displaying records.
281 ob.setVisible(b: true);
282 this.dispose();
283 }
285 private void btnibookActionPerformed(java.awt.event.ActionEvent evt) {
286 Issue ob = new Issue(); // Creates a new instance of the "Issue" class, representing the form for issuing a book.
287 ob.setVisible(b: true);
288 this.dispose();
289 }
291 private void btnrbookActionPerformed(java.awt.event.ActionEvent evt) {
292 Return ob = new Return(); // Creates a new instance of the "Return" class, representing the form for returning a book.
293 ob.setVisible(b: true);
294 this.dispose();
295 }
297 private void btnprintActionPerformed(java.awt.event.ActionEvent evt) {
298 Print ob = new Print(); // Creates a new instance of the "Print" class, representing the form for printing the library .
299 ob.setVisible(b: true);
300 this.dispose();
301 }
303 private void jMenu1ActionPerformed(java.awt.event.ActionEvent evt) {
304 // TODO add your handling code here:
305 }
Output
336:119
INS Unix (LF)
30°C
Clear
Search
2:32 AM
6/24/2023

Forgot Form:

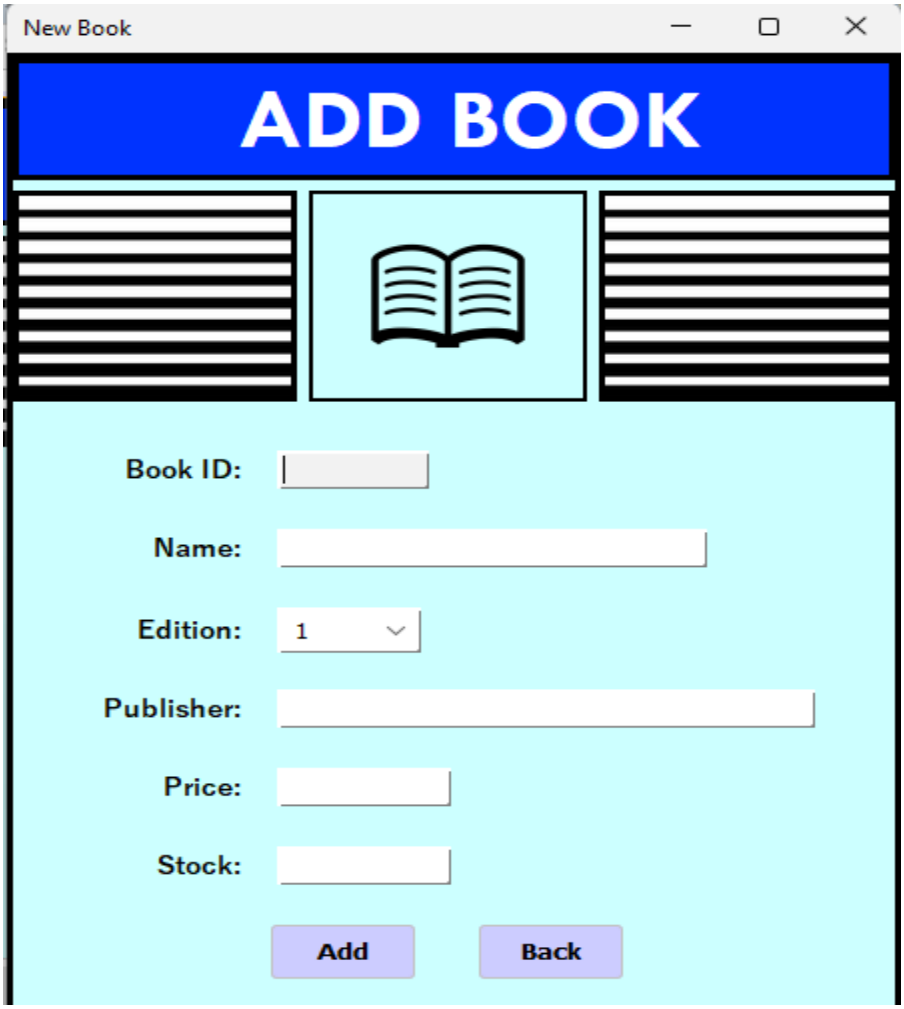

FORGOT PASSWORD
Username:
Search
Name:
Your Security Question:
Answer:
Get Password
Your Password is:
Back

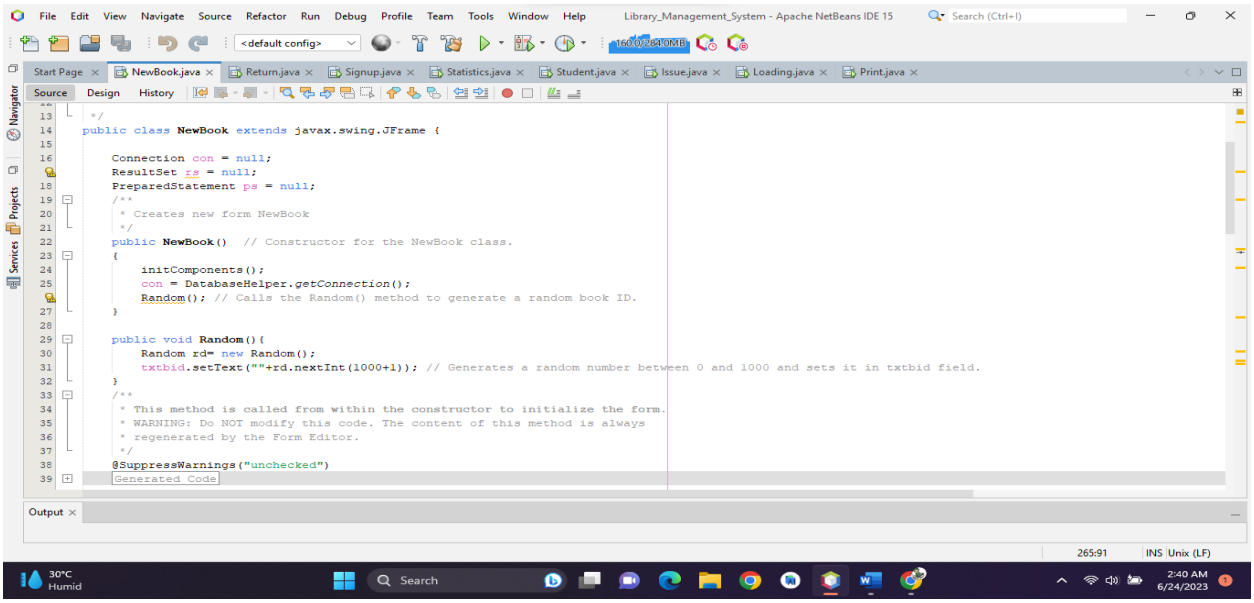

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Library_Management_System - Apache NetBeans IDE 15
Start Page Forgot.java NewBook.java Return.java Signup.java Statistics.java Student.java Issue.java Loading.java Print.java
Source Design History
public class Forgot extends javax.swing.JFrame {
Connection con = null;
ResultSet rs = null; // Declaration and initialization of variables for database operations.
PreparedStatement ps = null;
/**
* Creates new form Forgot
*/
public Forgot() // Constructor for the "Forgot" class.
{
initComponents();
con = DatabaseHelper.getConnection();
}
/**
* This method is called from within the constructor to initialize the form.
* WARNING: Do NOT modify this code. The content of this method is always
* regenerated by the Form Editor.
*/
@SuppressWarnings("unchecked")
Generated Code
private void btnbackActionPerformed(java.awt.event.ActionEvent evt) {
Login lo = new Login();
lo.setVisible(b: true);
this.dispose();
}
Output
317:109 INS Unix (LF)
30°C Clear
Search
2:38 AM 6/24/2023

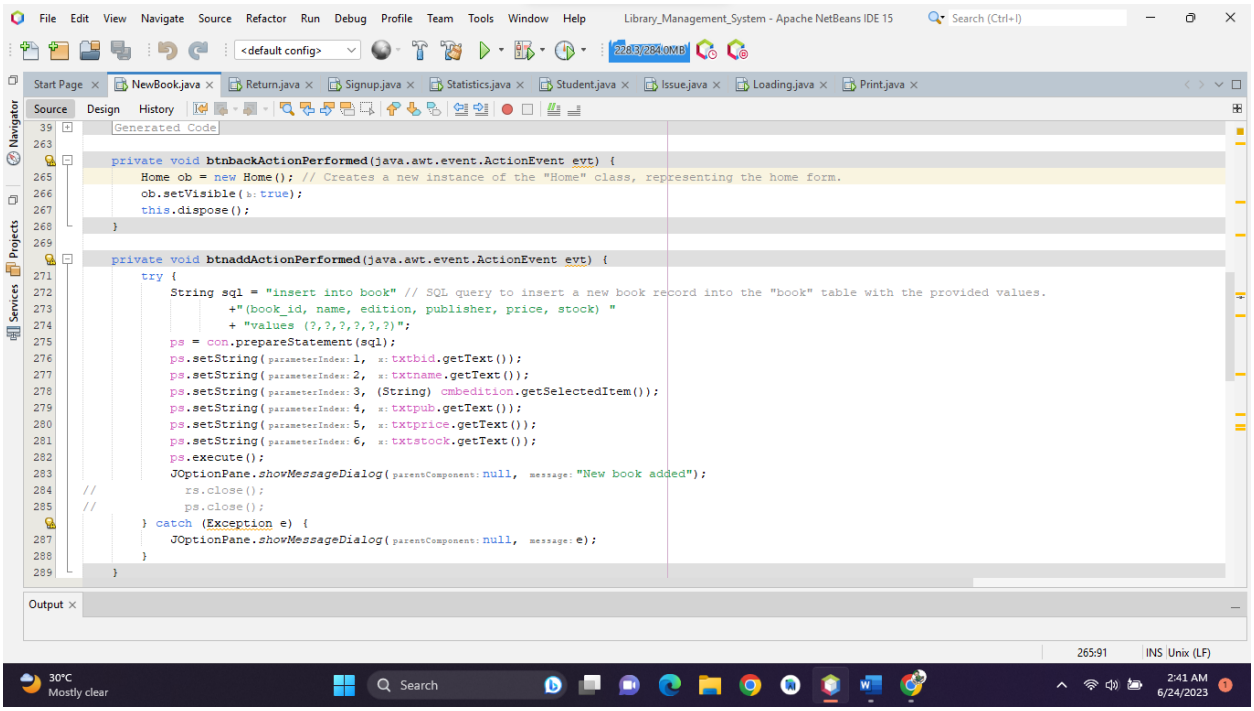

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Library_Management_System - Apache NetBeans IDE 15
Search (Ctrl+I)
<default config>
223.4/284.0MB
Start Page
Forgot.java
NewBook.java
Return.java
Signup.java
Statistics.java
Student.java
Issue.java
Loading.java
Print.java
Source
Design
History
Navigator
Projects
Services
private void btnsearchActionPerformed(java.awt.event.ActionEvent evt) {
search(); // Calls the search method to perform the search operation.
}
private void btnretriveActionPerformed(java.awt.event.ActionEvent evt) {
retrive(); // Calls the retrieve method to perform the search operation.
}
public void search() /* This method searches the database for a user account based on the entered username */ {
String a1= txtuname.getText(); // Takes the username from text user name field and stores it in String a1.
String sql = "select * from account where username='"+a1+"'"; // SQL query to select from the account table where the username matches the input.
try {
ps = con.prepareStatement(sql);
rs = ps.executeQuery();
if (rs.next()) {
txtname.setText(t: rs.getString(columnIndex: 2));
txtsques.setText(t: rs.getString(columnIndex: 4));
rs.close();
ps.close();
}else{
JOptionPane.showMessageDialog(parentComponent: null, message: "Incorrect Username"); // // If no matching record is found, it displays a "Incorrect
}
} catch (Exception e) {
JOptionPane.showMessageDialog(parentComponent: null, message: e); // Handles any exceptions that occurs during the operation.
}
}
Output
317:109
INS Unix (LF)
30°C
Mostly clear
Search
2:38 AM
6/24/2023


File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Library_Management_System - Apache NetBeans IDE 15
Search (Ctrl+I)
<default config>
230.0/284.0MB
Start Page
Forgot.java
NewBook.java
Return.java
Signup.java
Statistics.java
Student.java
Issue.java
Loading.java
Print.java
Source
Design
History
Navigator
Projects
Services
public void retrive() // This method retrieves the password from the database based on the provided security question answer.
{
String a1 = txtuname.getText();
String a2 = txtanswer.getText();
String sql = "select * from account where answer='"+a2+"'";
try {
ps = con.prepareStatement(sql);
rs = ps.executeQuery();
if (rs.next()) {
txtpassword.setText(t: rs.getString(columnIndex: 3));
// rs.close();
// ps.close();
}else{
JOptionPane.showMessageDialog(parentComponent: null, message: "Incorrect Answer to the security question!"); // If no matching record is found, it
}
} catch (Exception e) {
JOptionPane.showMessageDialog(parentComponent: null, message: e); // Handles any exceptions and displays an error message.
}
}
/**
* @param args the command line arguments
*/
public static void main(String args[]) {
/* Set the Nimbus look and feel */
Look and feel setting code (optional)
Output
317:109
INS Unix (LF)
30°C
Mostly clear
Search
2:39 AM
6/24/2023

New Book Form:

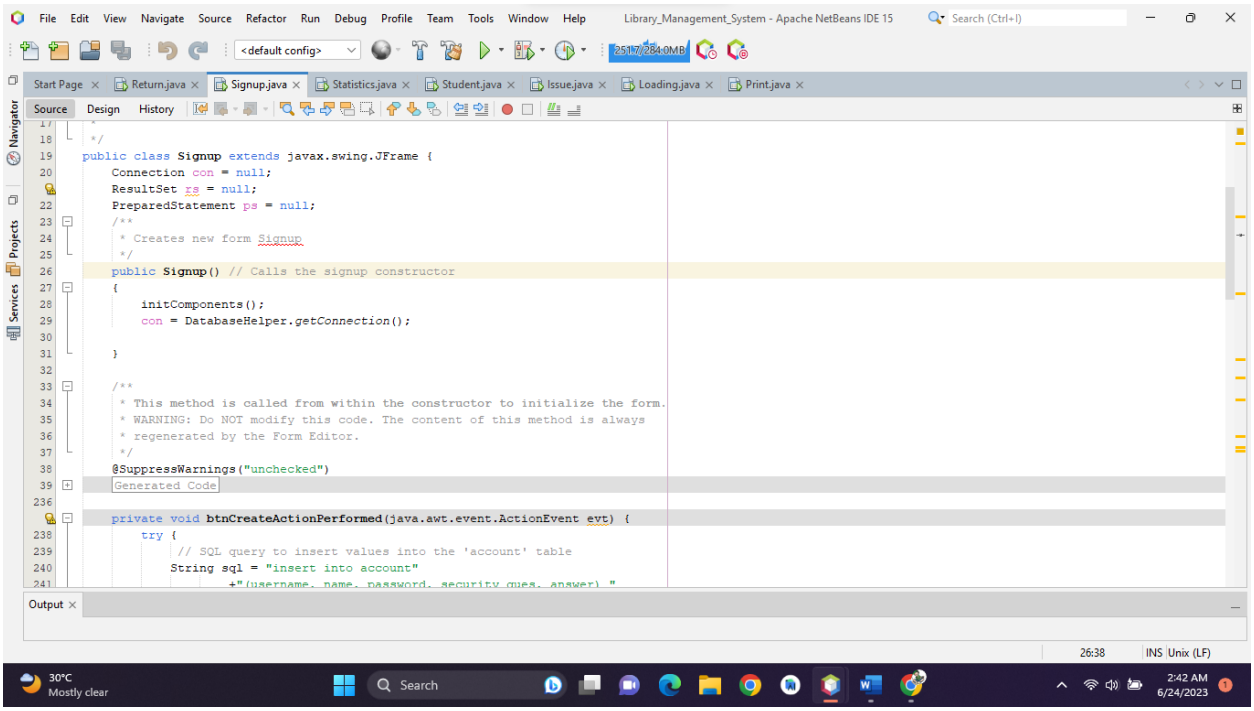


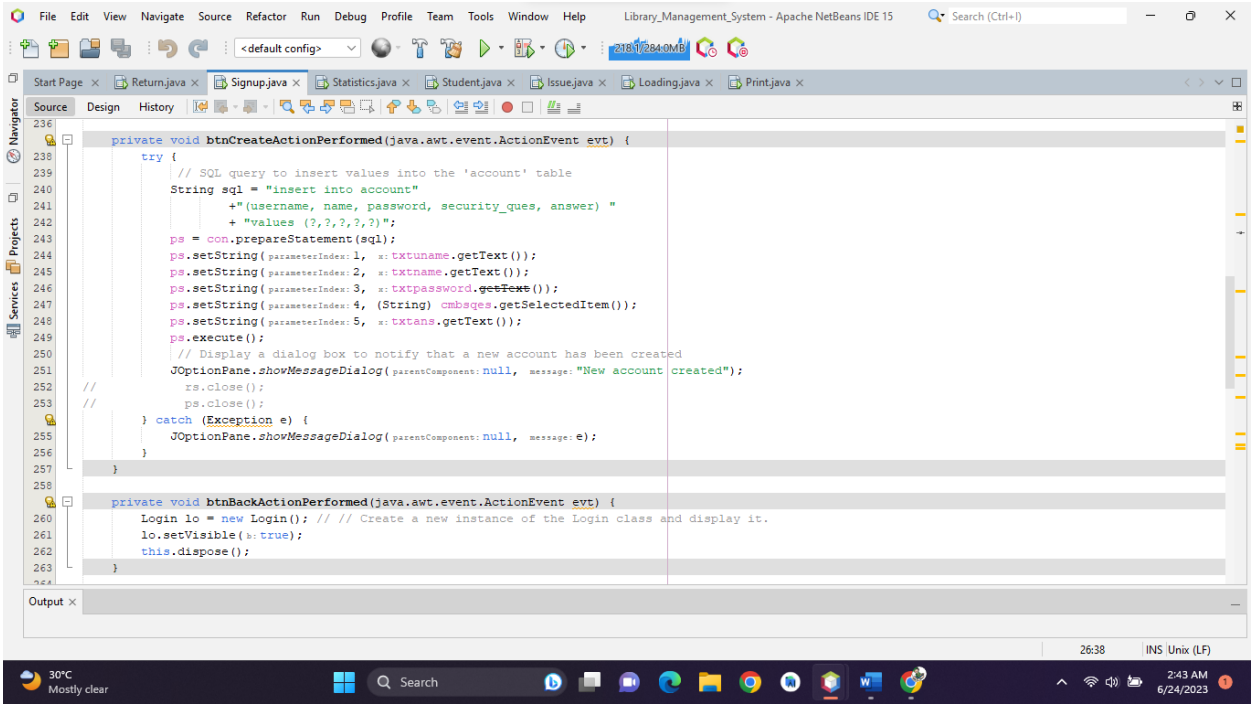



File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Library_Management_System - Apache NetBeans IDE 15
Start Page
NewBook.java
Return.java
Signup.java
Statistics.java
Student.java
Issue.java
Loading.java
Print.java
Source Design History
Generated Code
private void btnbackActionPerformed(java.awt.event.ActionEvent evt) {
Home ob = new Home(); // Creates a new instance of the "Home" class, representing the home form.
ob.setVisible(b: true);
this.dispose();
}
private void btnaddActionPerformed(java.awt.event.ActionEvent evt) {
try {
String sql = "insert into book" // SQL query to insert a new book record into the "book" table with the provided values.
+"(book_id, name, edition, publisher, price, stock) "
+ "values (?,?,?,?,?,?)";
ps = con.prepareStatement(sql);
ps.setString(parameterIndex: 1, x: txtbid.getText());
ps.setString(parameterIndex: 2, x: txtname.getText());
ps.setString(parameterIndex: 3, (String) cmbedition.getSelectedItem());
ps.setString(parameterIndex: 4, x: txtpub.getText());
ps.setString(parameterIndex: 5, x: txtprice.getText());
ps.setString(parameterIndex: 6, x: txtstock.getText());
ps.execute();
JOptionPane.showMessageDialog(parentComponent: null, message: "New book added");
// rs.close();
// ps.close();
} catch (Exception e) {
JOptionPane.showMessageDialog(parentComponent: null, message: e);
}
}
Output
265:91
INS Unix (LF)
30°C
Mostly clear
Search
2:41 AM
6/24/2023

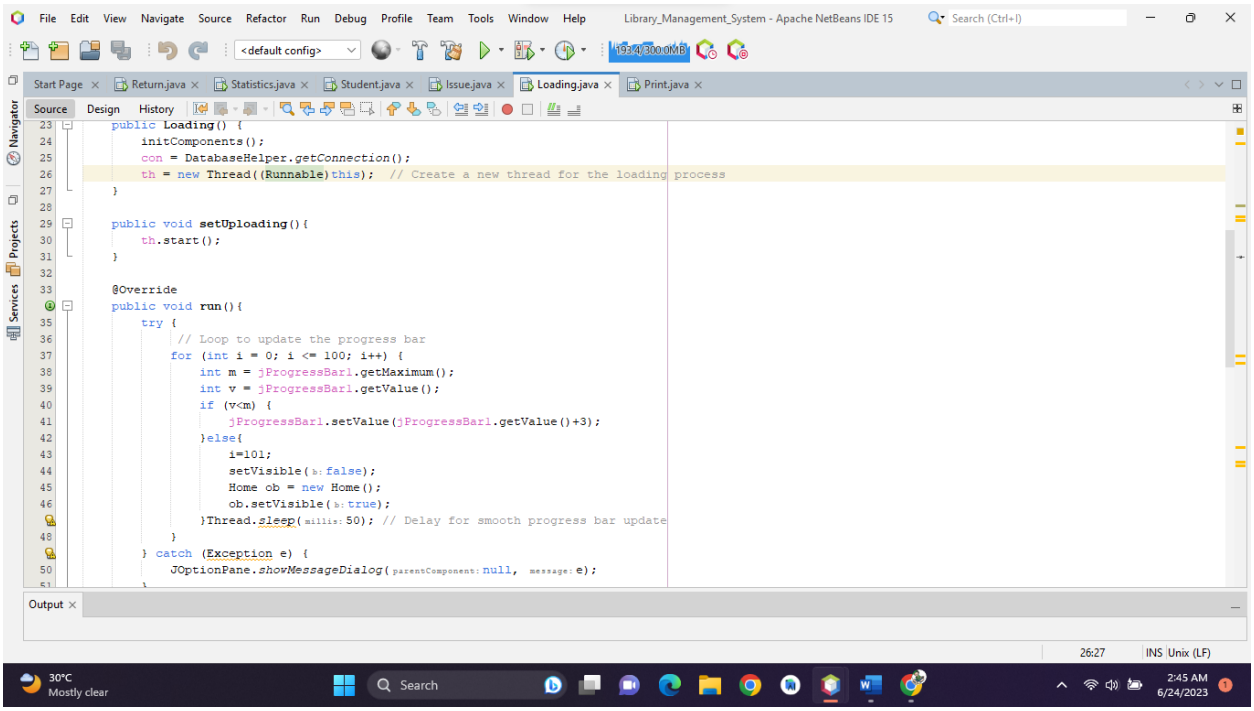
Signup Form:


New Account
REGISTER ACCOUNT
Username:
Name:
Password:
Security Question:
What is your mother tongue?
Answer:
Create
Back

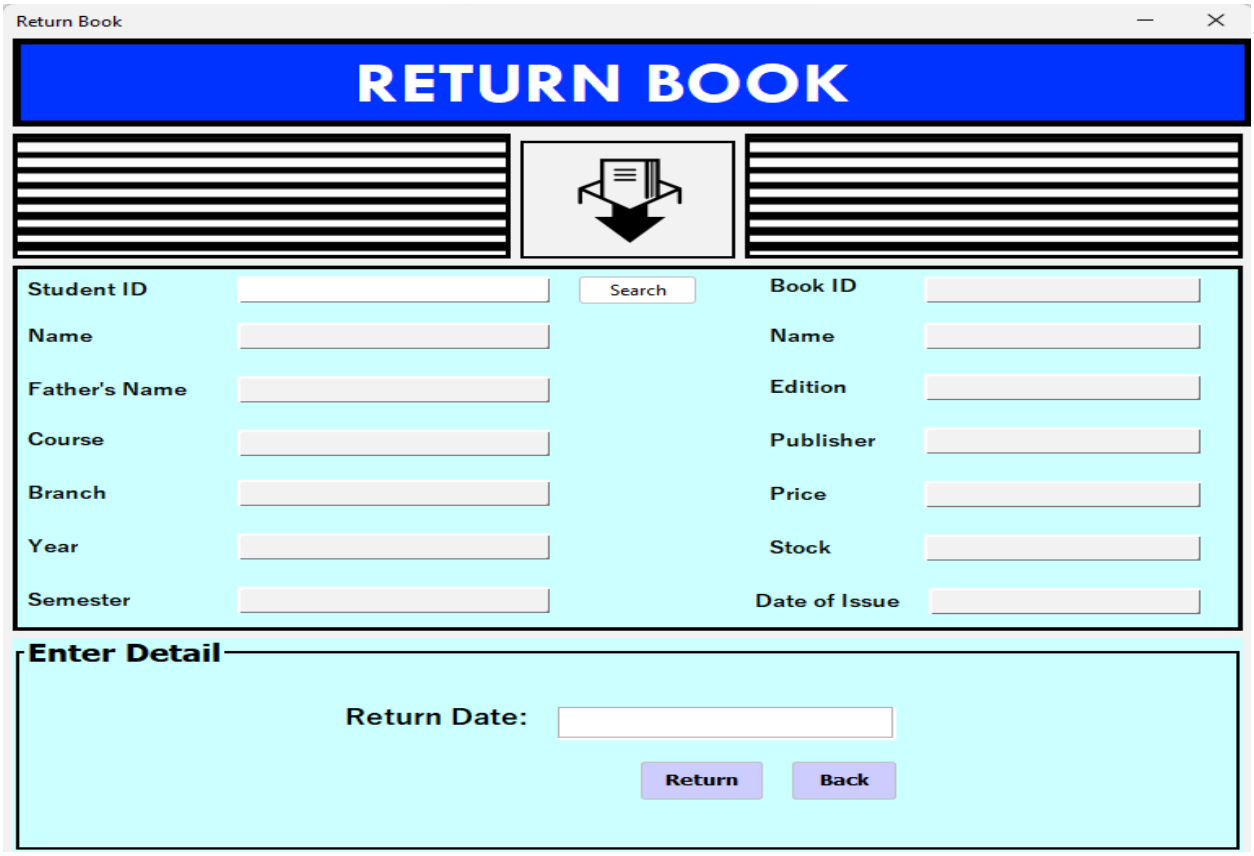

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Library_Management_System - Apache NetBeans IDE 15
Search (Ctrl+I)
<default config>
Start Page
Return.java
Signup.java
Statistics.java
Student.java
Issue.java
Loading.java
Print.java
Source
Design
History
Navigator
Projects
Services
*/
public class Signup extends javax.swing.JFrame {
Connection con = null;
ResultSet rs = null;
PreparedStatement ps = null;
/**
* Creates new form Signup
*/
public Signup() // Calls the signup constructor
{
initComponents();
con = DatabaseHelper.getConnection();
}
/**
* This method is called from within the constructor to initialize the form.
* WARNING: Do NOT modify this code. The content of this method is always
* regenerated by the Form Editor.
*/
@SuppressWarnings("unchecked")
Generated Code
private void btnCreateActionPerformed(java.awt.event.ActionEvent evt) {
try {
// SQL query to insert values into the 'account' table
String sql = "insert into account"
+"(username, name, password, security_ques, answer) "
Output
26:38
INS Unix (LF)
30°C
Mostly clear
Search
2:42 AM
6/24/2023

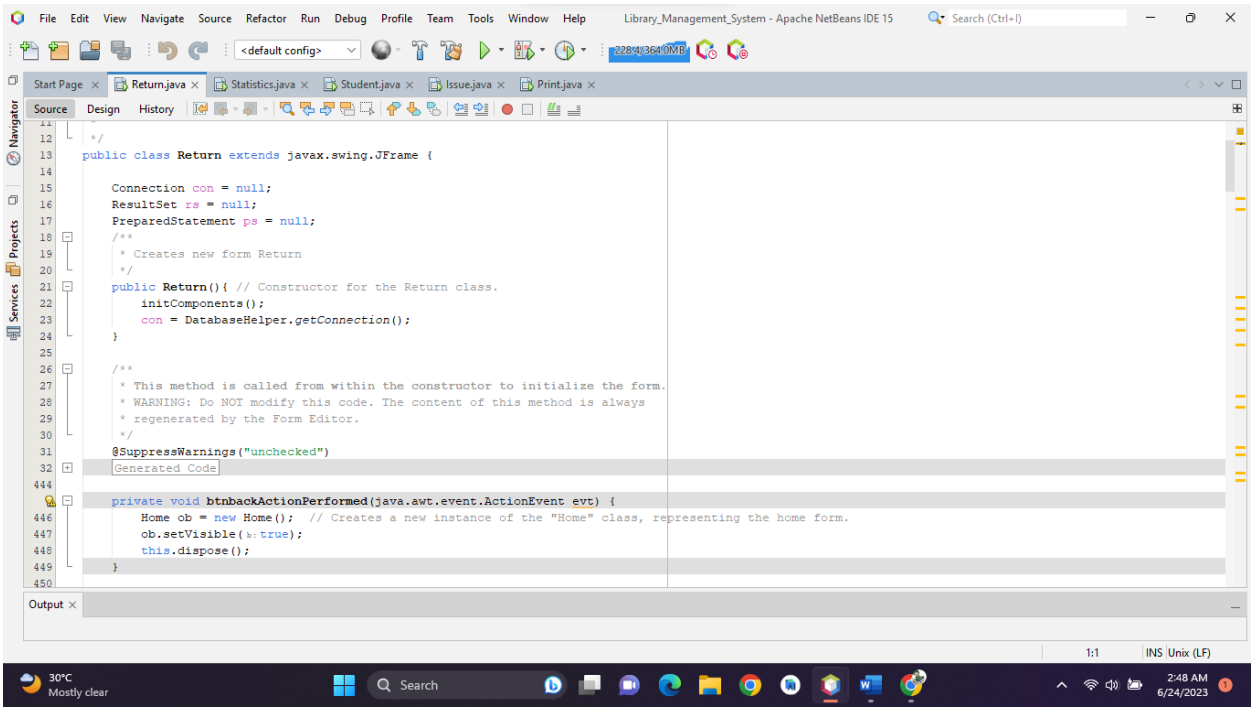

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Library_Management_System - Apache NetBeans IDE 15
Search (Ctrl+I)
<default config>
Start Page
Return.java
Signup.java
Statistics.java
Student.java
Issue.java
Loading.java
Print.java
Source
Design
History
Navigator
Projects
Services
private void btnCreateActionPerformed(java.awt.event.ActionEvent evt) {
try {
// SQL query to insert values into the 'account' table
String sql = "insert into account"
+"(username, name, password, security_ques, answer) "
+ "values (?,?,?,?,?)";
ps = con.prepareStatement(sql);
ps.setString(parameterIndex: 1, x: txtuname.getText());
ps.setString(parameterIndex: 2, x: txtname.getText());
ps.setString(parameterIndex: 3, x: txtpassword.getText());
ps.setString(parameterIndex: 4, (String) cmbsqes.getSelectedItem());
ps.setString(parameterIndex: 5, x: txtans.getText());
ps.execute();
// Display a dialog box to notify that a new account has been created
JOptionPane.showMessageDialog(parentComponent: null, message: "New account created");
// rs.close();
// ps.close();
} catch (Exception e) {
JOptionPane.showMessageDialog(parentComponent: null, message: e);
}
}
private void btnBackActionPerformed(java.awt.event.ActionEvent evt) {
Login lo = new Login(); // // Create a new instance of the Login class and display it.
lo.setVisible(b: true);
this.dispose();
}
Output
26:38
INS Unix (LF)
30°C
Mostly clear
Search
2:43 AM
6/24/2023

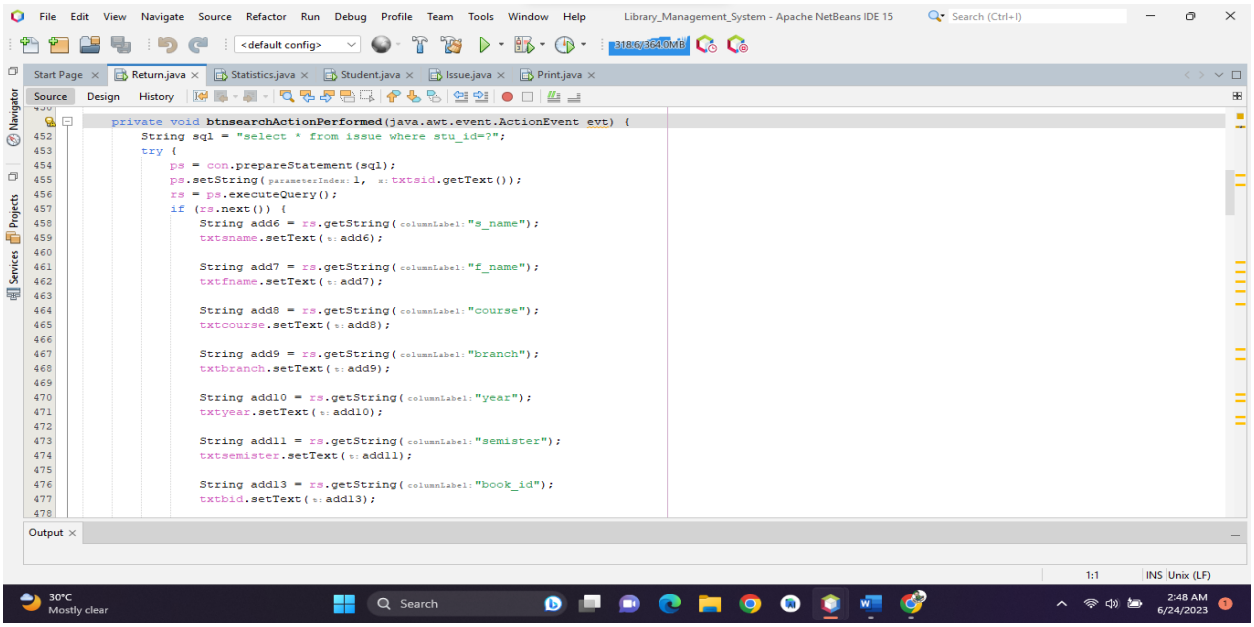
Loading Form:



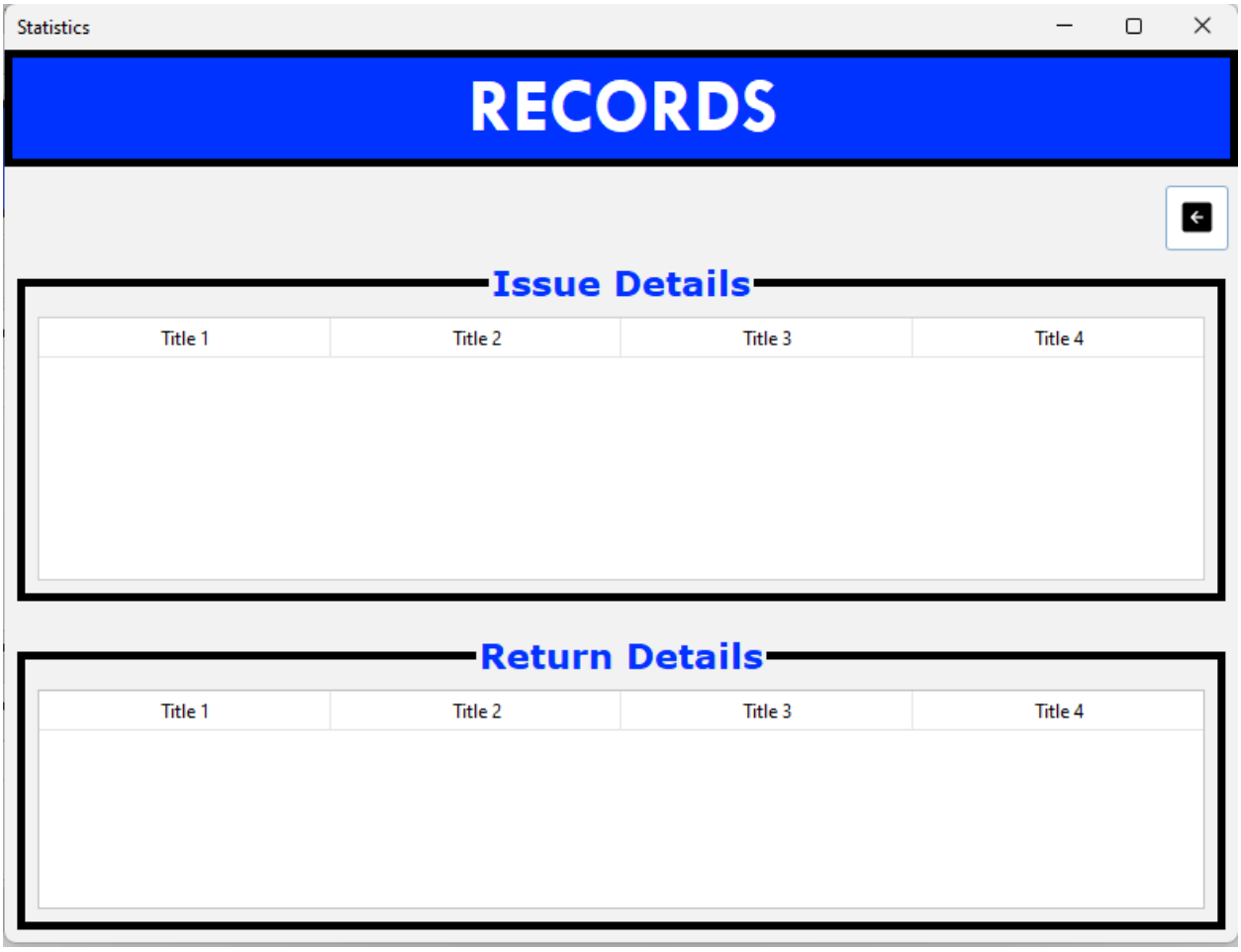
Return Book Form:

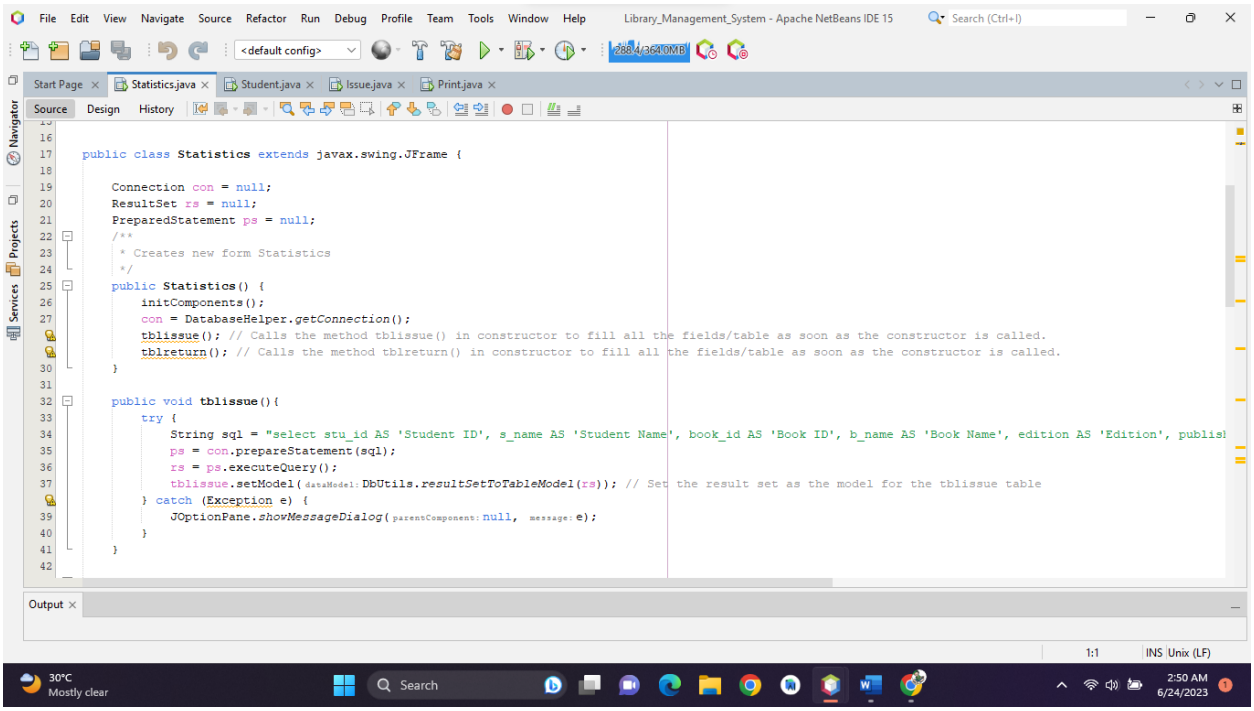


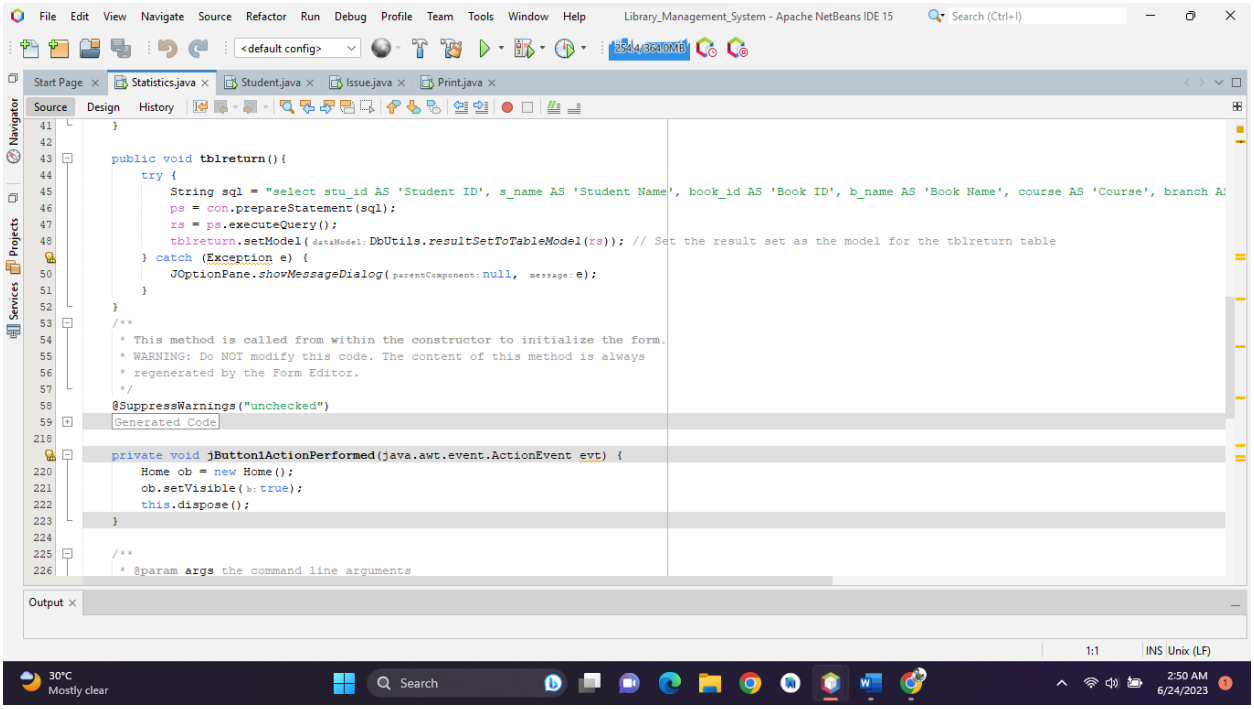




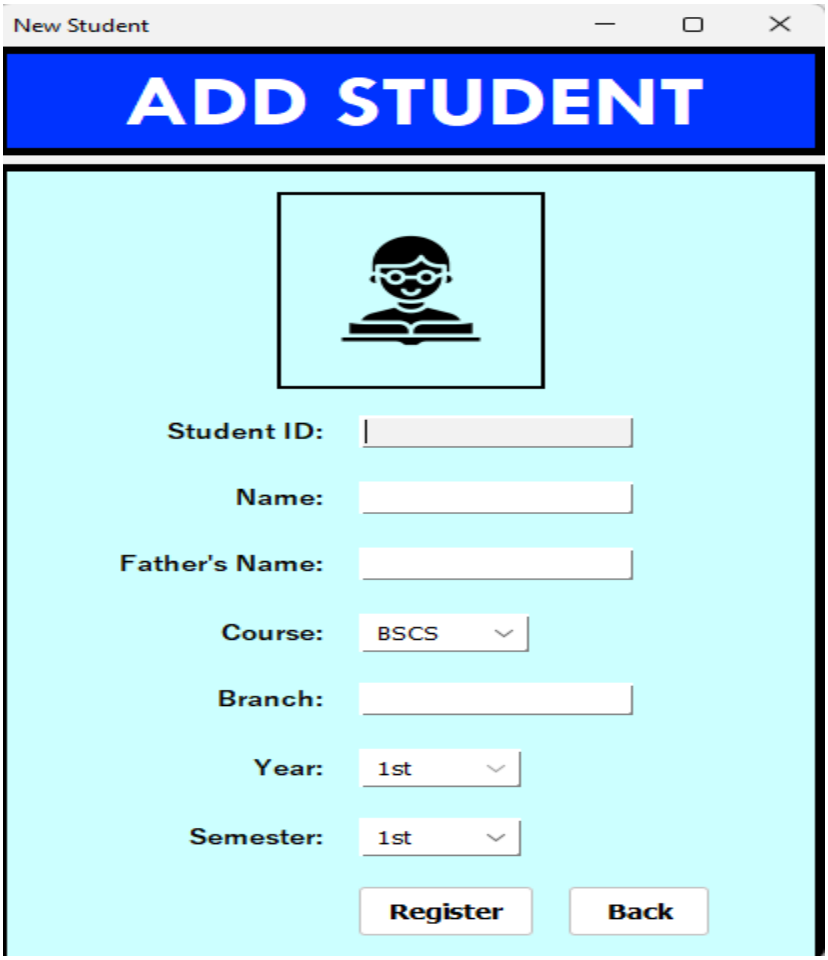
Statistics(Records) Form:

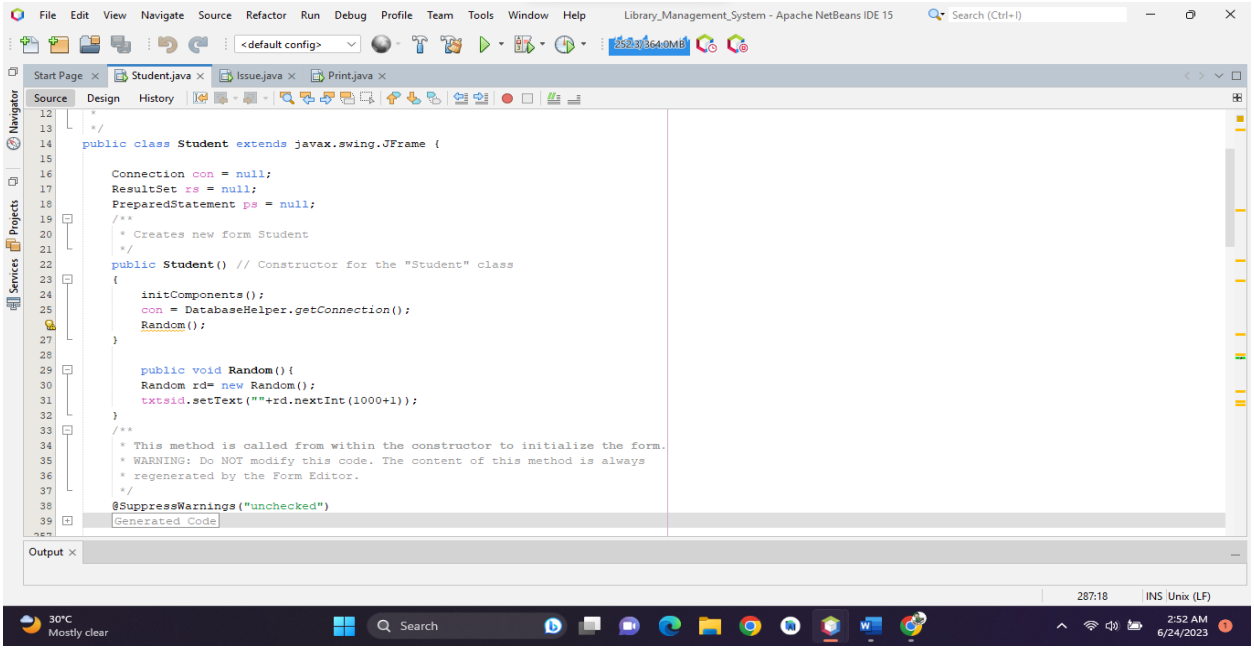


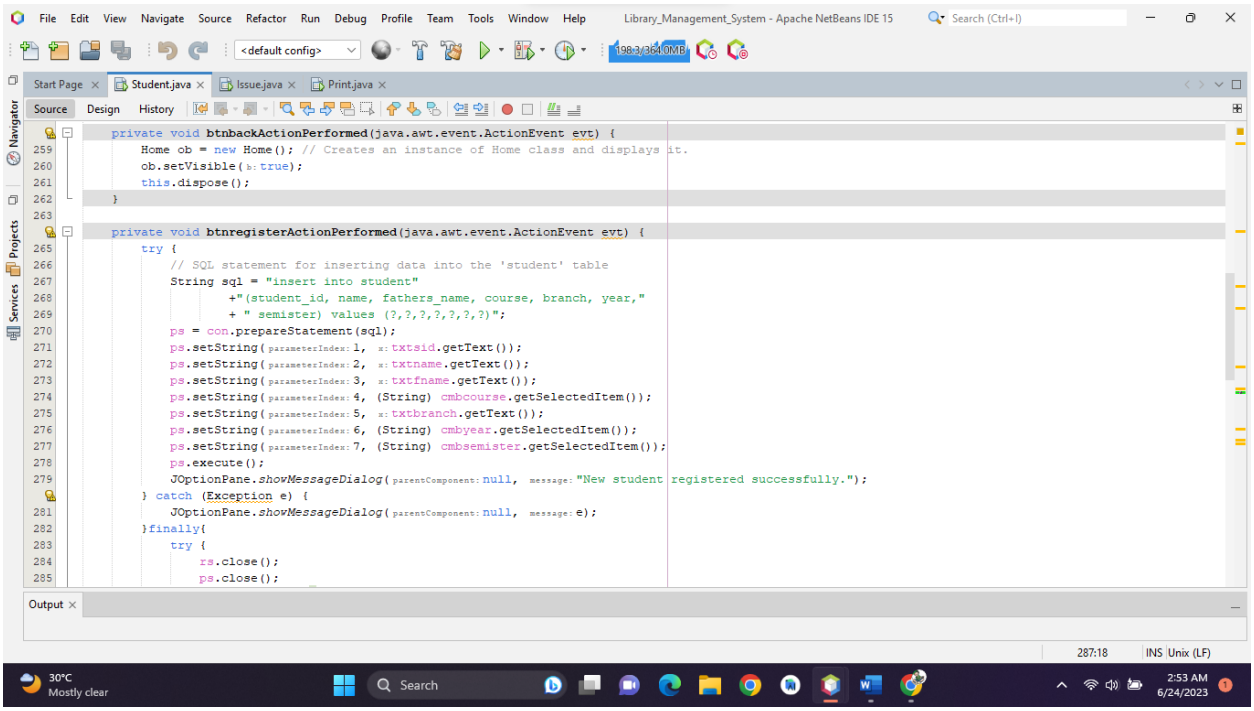

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Library_Management_System - Apache NetBeans IDE 15
Search (Ctrl+I)
<default config>
Start Page
Statistics.java
Student.java
Issue.java
Print.java
Source
Design
History
public class Statistics extends javax.swing.JFrame {
Connection con = null;
ResultSet rs = null;
PreparedStatement ps = null;
/**
* Creates new form Statistics
*/
public Statistics() {
initComponents();
con = DatabaseHelper.getConnection();
tblissue(); // Calls the method tblissue() in constructor to fill all the fields/table as soon as the constructor is called.
tblreturn(); // Calls the method tblreturn() in constructor to fill all the fields/table as soon as the constructor is called.
}
public void tblissue(){
try {
String sql = "select stu_id AS 'Student ID', s_name AS 'Student Name', book_id AS 'Book ID', b_name AS 'Book Name', edition AS 'Edition', publis
ps = con.prepareStatement(sql);
rs = ps.executeQuery();
tblissue.setModel(dataModel: DbUtils.resultSetToTableModel(rs)); // Set the result set as the model for the tblissue table
} catch (Exception e) {
JOptionPane.showMessageDialog(parentComponent: null, message: e);
}
}
Output
1:1
INS Unix (LF)
30°C
Mostly clear
Search
2:50 AM
6/24/2023


File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Library_Management_System - Apache NetBeans IDE 15
Search (Ctrl+I)
<default config>
Start Page
Statistics.java
Student.java
Issue.java
Print.java
Source
Design
History
}
public void tblreturn(){
try {
String sql = "select stu_id AS 'Student ID', s_name AS 'Student Name', book_id AS 'Book ID', b_name AS 'Book Name', course AS 'Course', branch A
ps = con.prepareStatement(sql);
rs = ps.executeQuery();
tblreturn.setModel(dataModel: DbUtils.resultSetToTableModel(rs)); // Set the result set as the model for the tblreturn table
} catch (Exception e) {
JOptionPane.showMessageDialog(parentComponent: null, message: e);
}
}
/**
* This method is called from within the constructor to initialize the form.
* WARNING: Do NOT modify this code. The content of this method is always
* regenerated by the Form Editor.
*/
@SuppressWarnings("unchecked")
Generated Code
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
Home ob = new Home();
ob.setVisible(b: true);
this.dispose();
}
/**
* @param args the command line arguments
Output
1:1
INS Unix (LF)
30°C
Mostly clear
Search
2:50 AM
6/24/2023

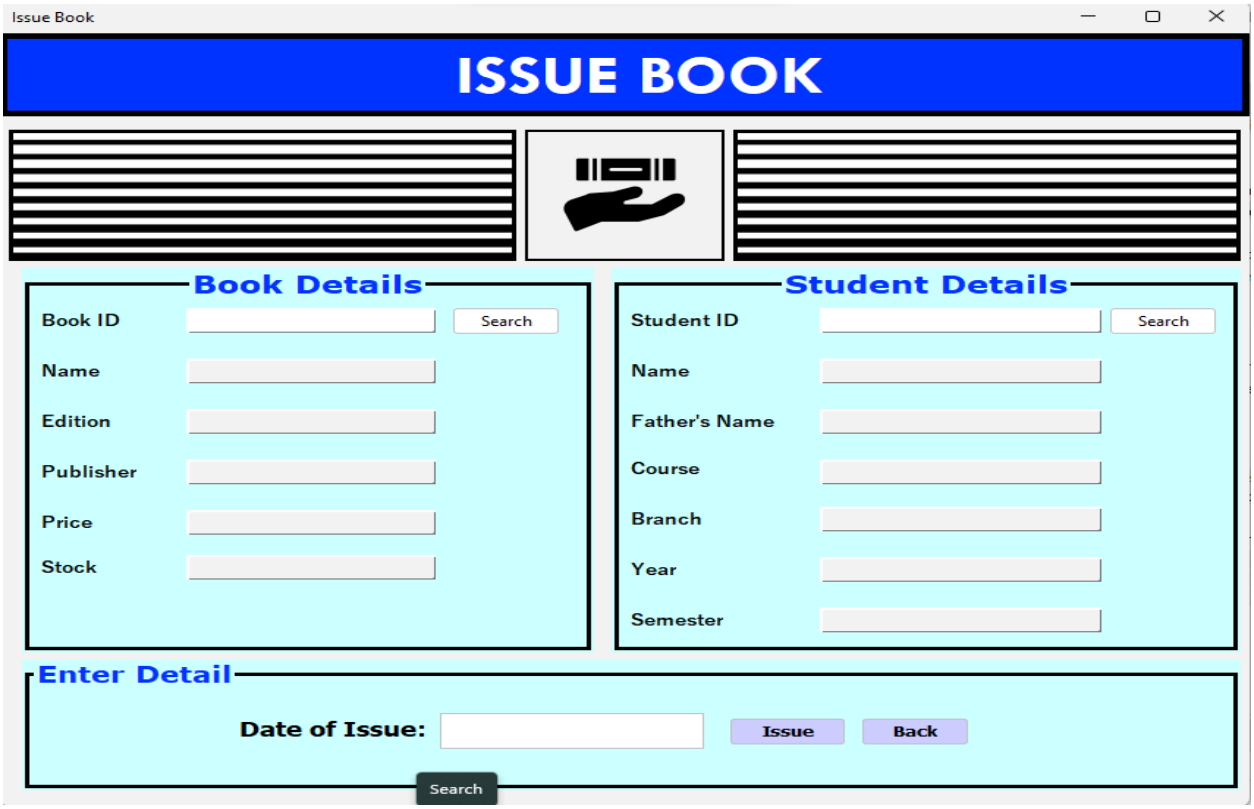
Add Student Form:

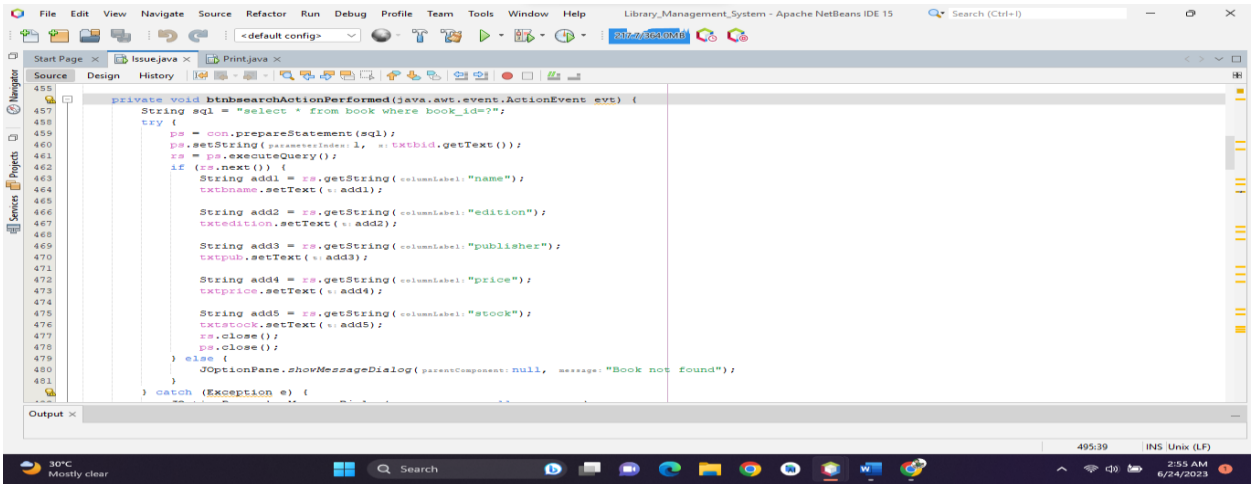




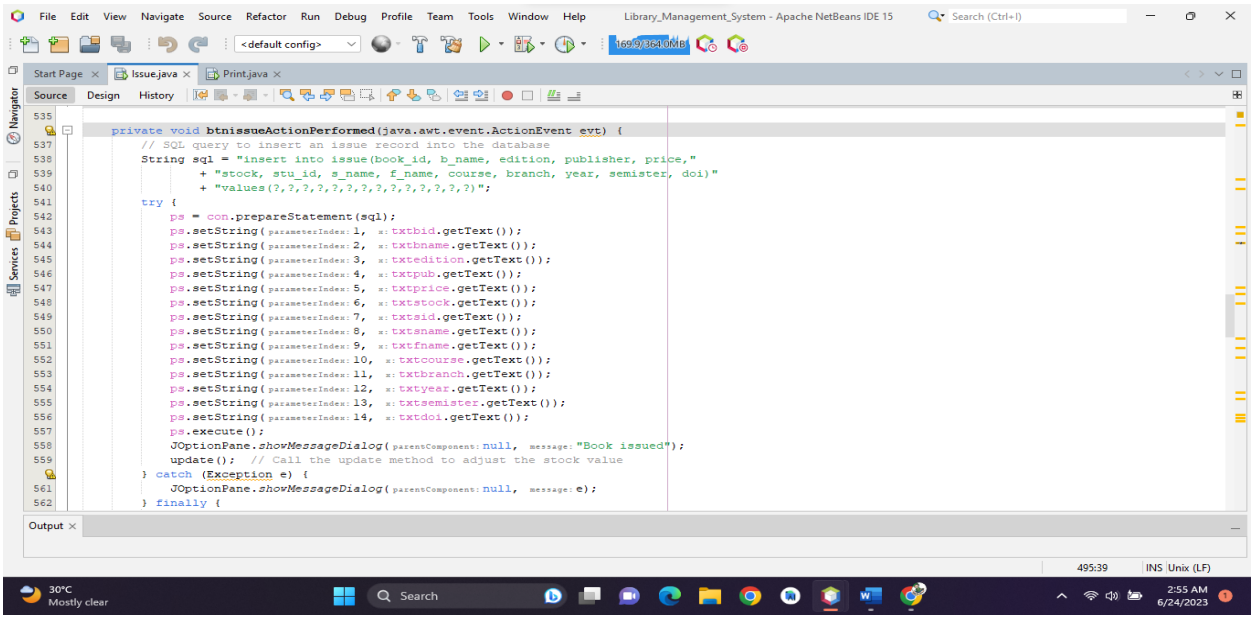

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
Library_Management_System - Apache NetBeans IDE 15
Start Page
Student.java
Issue.java
Print.java
Source
Design
History
private void btnbackActionPerformed(java.awt.event.ActionEvent evt) {
Home ob = new Home(); // Creates an instance of Home class and displays it.
ob.setVisible(b: true);
this.dispose();
}
private void btnregisterActionPerformed(java.awt.event.ActionEvent evt) {
try {
// SQL statement for inserting data into the 'student' table
String sql = "insert into student"
+"(student_id, name, fathers_name, course, branch, year,"
+ " semister) values (?,?,?,?,?,?,?)";
ps = con.prepareStatement(sql);
ps.setString(parameterIndex: 1, x: txtsid.getText());
ps.setString(parameterIndex: 2, x: txtname.getText());
ps.setString(parameterIndex: 3, x: txtfname.getText());
ps.setString(parameterIndex: 4, (String) cmbcourse.getSelectedItem());
ps.setString(parameterIndex: 5, x: txtbranch.getText());
ps.setString(parameterIndex: 6, (String) cmbyear.getSelectedItem());
ps.setString(parameterIndex: 7, (String) cmbsemister.getSelectedItem());
ps.execute();
JOptionPane.showMessageDialog(parentComponent: null, message: "New student registered successfully.");
} catch (Exception e) {
JOptionPane.showMessageDialog(parentComponent: null, message: e);
}finally{
try {
rs.close();
ps.close();
Output
287:18
INS Unix (LF)
30°C
Mostly clear
Search
2:53 AM
6/24/2023

Issue Form:

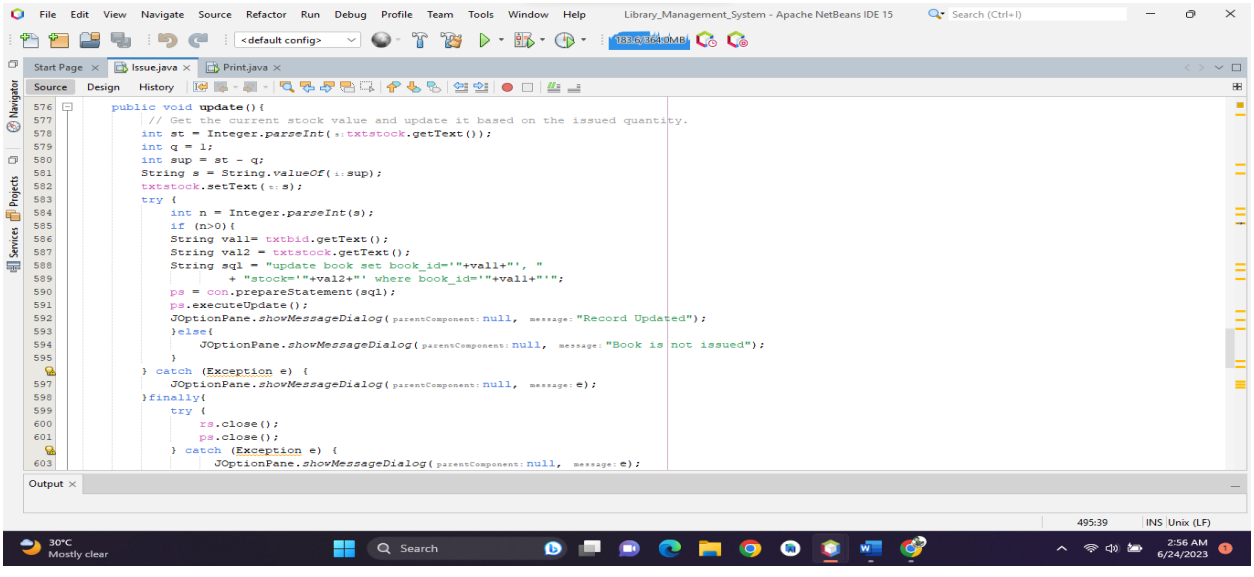

Issue Book
ISSUE BOOK
Book Details
Book ID
Search
Name
Edition
Publisher
Price
Stock
Student Details
Student ID
Search
Name
Father's Name
Course
Branch
Year
Semester
Enter Detail
Date of Issue:
Issue
Back
Search



```
455
456
457     private void btnbsearchActionPerformed(java.awt.event.ActionEvent evt) {
458         String sql = "select * from book where book_id=?";
459         try {
460             ps = con.prepareStatement(sql);
461             ps.setString(parameterIndex: 1, x: txtbid.getText());
462             rs = ps.executeQuery();
463             if (rs.next()) {
464                 String add1 = rs.getString(columnLabel: "name");
465                 txtbname.setText(s: add1);
466
467                 String add2 = rs.getString(columnLabel: "edition");
468                 txtedition.setText(s: add2);
469
470                 String add3 = rs.getString(columnLabel: "publisher");
471                 txtpub.setText(s: add3);
472
473                 String add4 = rs.getString(columnLabel: "price");
474                 txtprice.setText(s: add4);
475
476                 String add5 = rs.getString(columnLabel: "stock");
477                 txtstock.setText(s: add5);
478                 rs.close();
479                 ps.close();
480             } else {
481                 JOptionPane.showMessageDialog(parentComponent: null, message: "Book not found");
482             }
483         } catch (Exception e) {
```

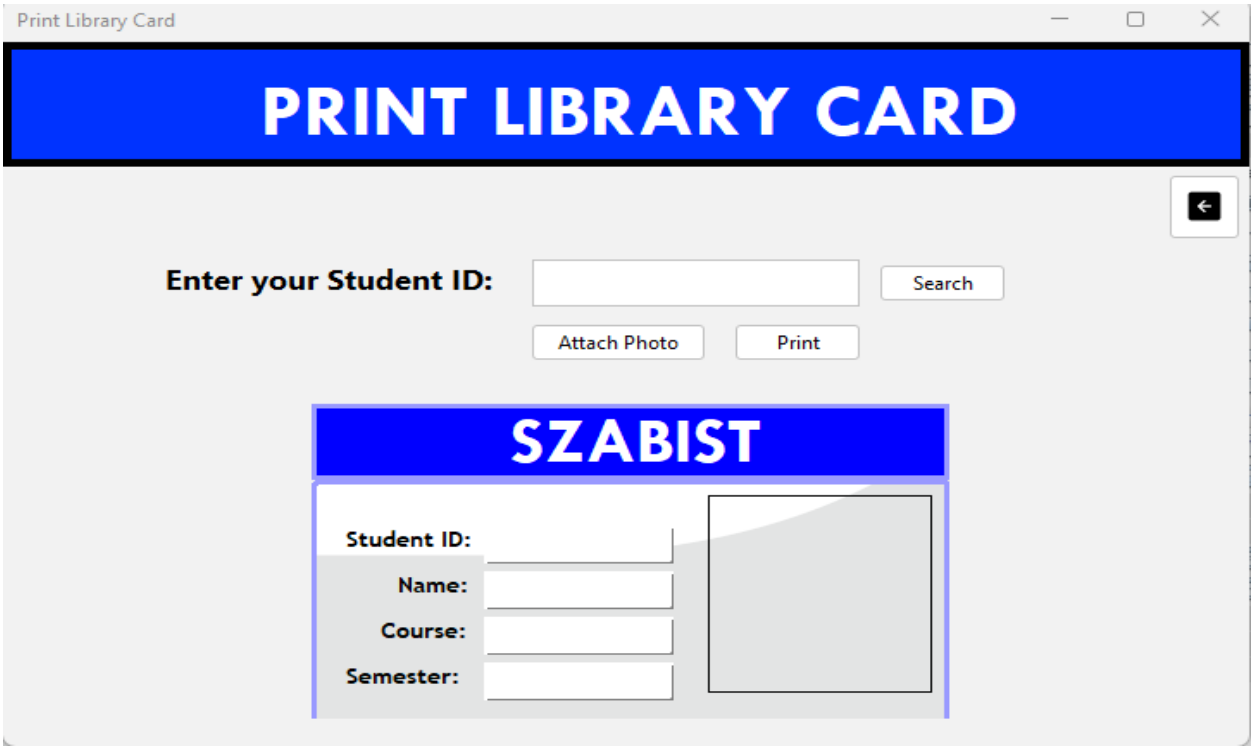


```
535
536     private void btnissueActionPerformed(java.awt.event.ActionEvent evt) {
537         // SQL query to insert an issue record into the database
538         String sql = "insert into issue(book_id, b_name, edition, publisher, price,"
539                 + "stock, stu_id, s_name, f_name, course, branch, year, semister, doi)"
540                 + "values(?,?,?,?,?,?,?,?,?,?,?,?,?,?)";
541         try {
542             ps = con.prepareStatement(sql);
543             ps.setString(parameterIndex: 1, x: txtbid.getText());
544             ps.setString(parameterIndex: 2, x: txtbname.getText());
545             ps.setString(parameterIndex: 3, x: txtedition.getText());
546             ps.setString(parameterIndex: 4, x: txtpub.getText());
547             ps.setString(parameterIndex: 5, x: txtprice.getText());
548             ps.setString(parameterIndex: 6, x: txtstock.getText());
549             ps.setString(parameterIndex: 7, x: txtsid.getText());
550             ps.setString(parameterIndex: 8, x: txtsname.getText());
551             ps.setString(parameterIndex: 9, x: txtfname.getText());
552             ps.setString(parameterIndex: 10, x: txtcourse.getText());
553             ps.setString(parameterIndex: 11, x: txtbranch.getText());
554             ps.setString(parameterIndex: 12, x: txtyear.getText());
555             ps.setString(parameterIndex: 13, x: txtsemister.getText());
556             ps.setString(parameterIndex: 14, x: txtdoi.getText());
557             ps.execute();
558             JOptionPane.showMessageDialog(parentComponent: null, message: "Book issued");
559             update();  // Call the update method to adjust the stock value
560         } catch (Exception e) {
561             JOptionPane.showMessageDialog(parentComponent: null, message: e);
562         } finally {
```

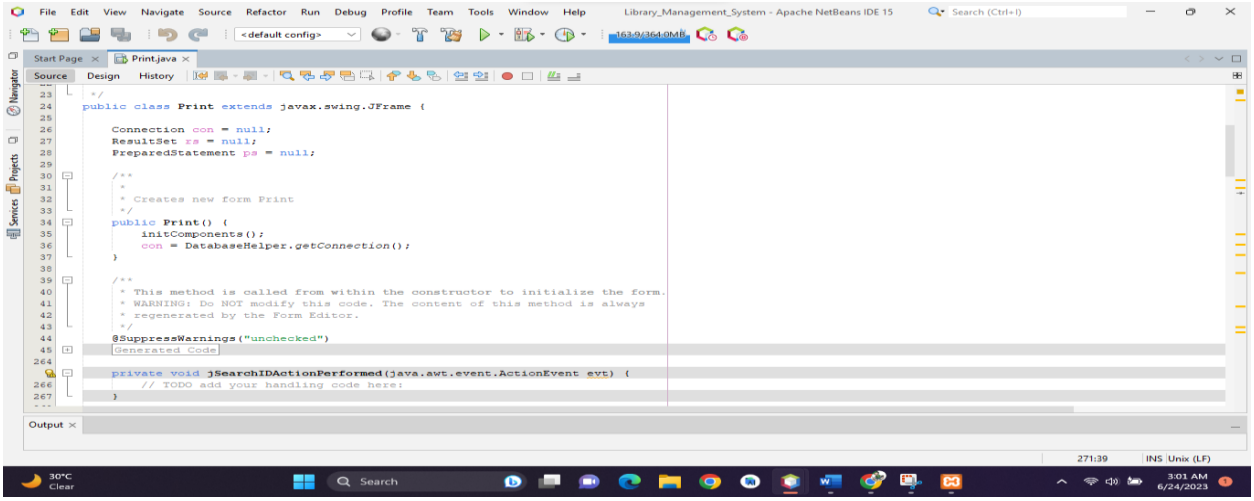


```
576     public void update(){
577         // Get the current stock value and update it based on the issued quantity.
578         int st = Integer.parseInt(s: txtstock.getText());
579         int q = 1;
580         int sup = st - q;
581         String s = String.valueOf(i: sup);
582         txtstock.setText(s: s);
583         try {
584             int n = Integer.parseInt(s);
585             if (n>0){
586             String val1= txtbid.getText();
587             String val2 = txtstock.getText();
588             String sql = "update book set book_id='"+val1+"', "
589                     + "stock='"+val2+"' where book_id='"+val1+"'";
590             ps = con.prepareStatement(sql);
591             ps.executeUpdate();
592             JOptionPane.showMessageDialog(parentComponent: null, message: "Record Updated");
593             }else{
594                 JOptionPane.showMessageDialog(parentComponent: null, message: "Book is not issued");
595             }
596         } catch (Exception e) {
597             JOptionPane.showMessageDialog(parentComponent: null, message: e);
598         }finally{
599             try {
600                 rs.close();
601                 ps.close();
602             } catch (Exception e) {
603                 JOptionPane.showMessageDialog(parentComponent: null, message: e);
```

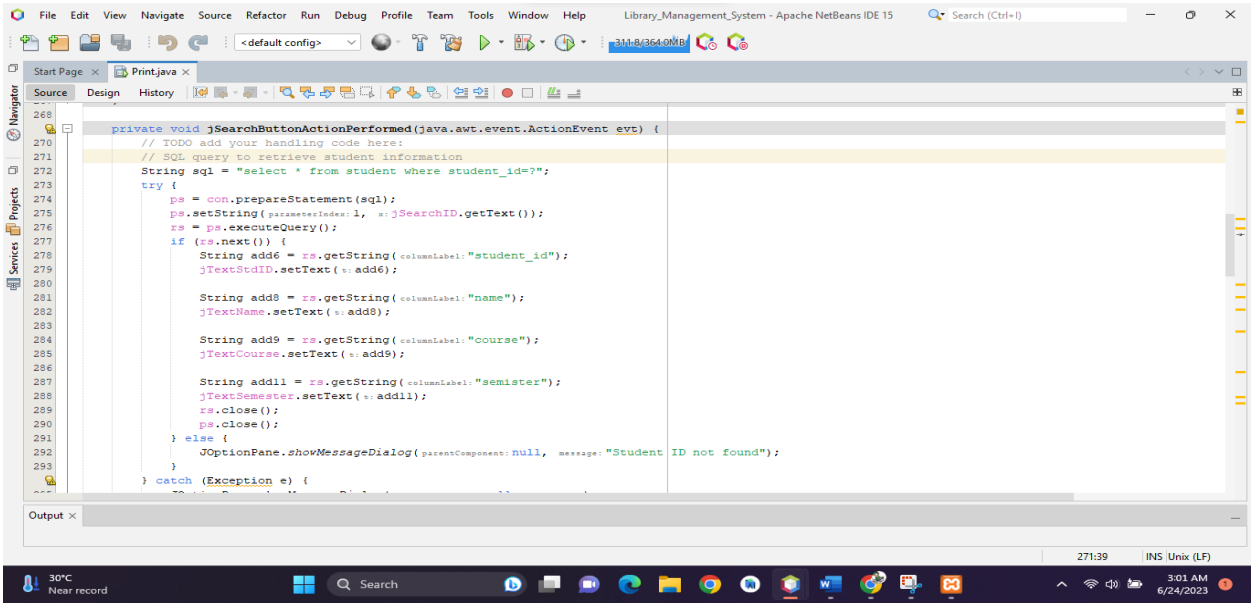
Print Form:


Print Library Card
PRINT LIBRARY CARD
Enter your Student ID:
Search
Attach Photo
Print
SZABIST
Student ID:
Name:
Course:
Semester:

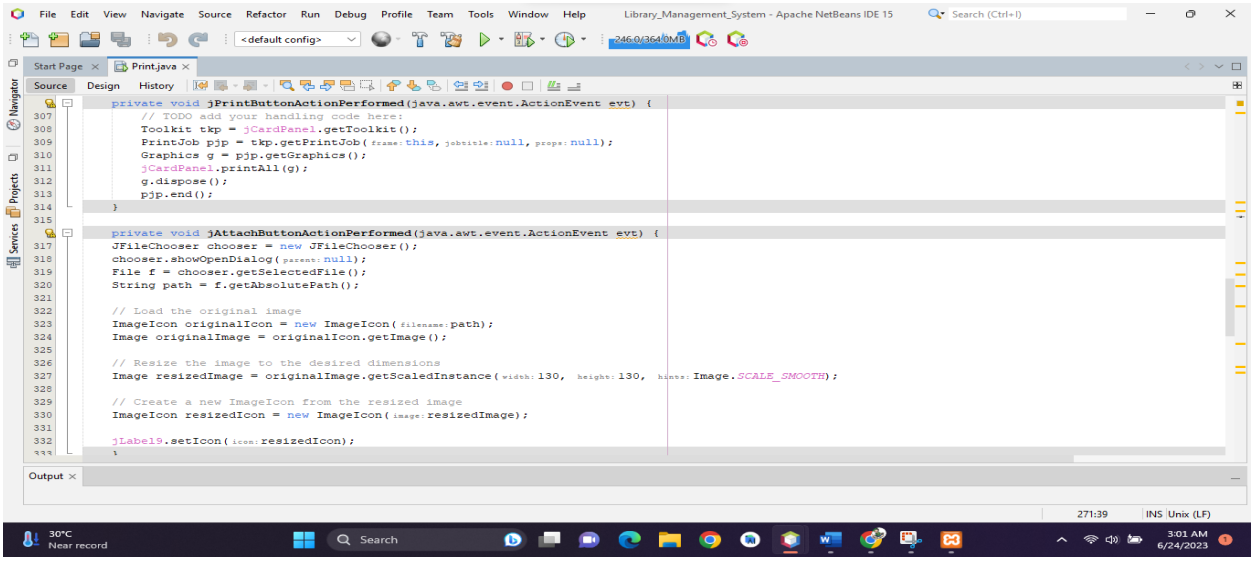

SZABIST
Student ID: 749
Name: Ilyaan
Course: BSc
Semester: 2nd



```
/**
 *
 * Creates new form Print
 */
public Print() {
    initComponents();
    con = DatabaseHelper.getConnection();
}

/**
 * This method is called from within the constructor to initialize the form.
 * WARNING: Do NOT modify this code. The content of this method is always
 * regenerated by the Form Editor.
 */
@SuppressWarnings("unchecked")
Generated Code

private void jSearchIDActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
}
```



```
private void jSearchButtonActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    // SQL query to retrieve student information
    String sql = "select * from student where student_id=?";
    try {
        ps = con.prepareStatement(sql);
        ps.setString(parameterIndex: 1, x: jSearchID.getText());
        rs = ps.executeQuery();
        if (rs.next()) {
            String add6 = rs.getString(columnLabel: "student_id");
            jTextStdID.setText(s: add6);

            String add8 = rs.getString(columnLabel: "name");
            jTextName.setText(s: add8);

            String add9 = rs.getString(columnLabel: "Course");
            jTextCourse.setText(s: add9);

            String add11 = rs.getString(columnLabel: "semister");
            jTextSemester.setText(s: add11);
            rs.close();
            ps.close();
        } else {
            JOptionPane.showMessageDialog(parentComponent: null, message: "Student ID not found");
        }
    } catch (Exception e) {
```



```
private void jPrintButtonActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    Toolkit tkp = jCardPanel.getToolkit();
    PrintJob pjp = tkp.getPrintJob(frame: this, jobtitle: null, props: null);
    Graphics g = pjp.getGraphics();
    jCardPanel.printAll(g);
    g.dispose();
    pjp.end();
}

private void jAttachButtonActionPerformed(java.awt.event.ActionEvent evt) {
JFileChooser chooser = new JFileChooser();
chooser.showOpenDialog(parent: null);
File f = chooser.getSelectedFile();
String path = f.getAbsolutePath();

// Load the original image
ImageIcon originalIcon = new ImageIcon(filename: path);
Image originalImage = originalIcon.getImage();

// Resize the image to the desired dimensions
Image resizedImage = originalImage.getScaledInstance(width: 130, height: 130, hints: Image.SCALE_SMOOTH);

// Create a new ImageIcon from the resized image
ImageIcon resizedIcon = new ImageIcon(image: resizedImage);

jLabel9.setIcon(icon: resizedIcon);
```

CONCLUSION

The Library Management System project successfully applies object-oriented programming principles to create an efficient and user-friendly system for managing library operations. It incorporates key OOP concepts, offers essential features, prioritizes data security, and provides a valuable learning experience. With potential for future enhancements, the system contributes to improved library management and user experience. The modular design and user-friendly interface of the Library Management System makes it easy for users to perform library related tasks.

THANK YOU
