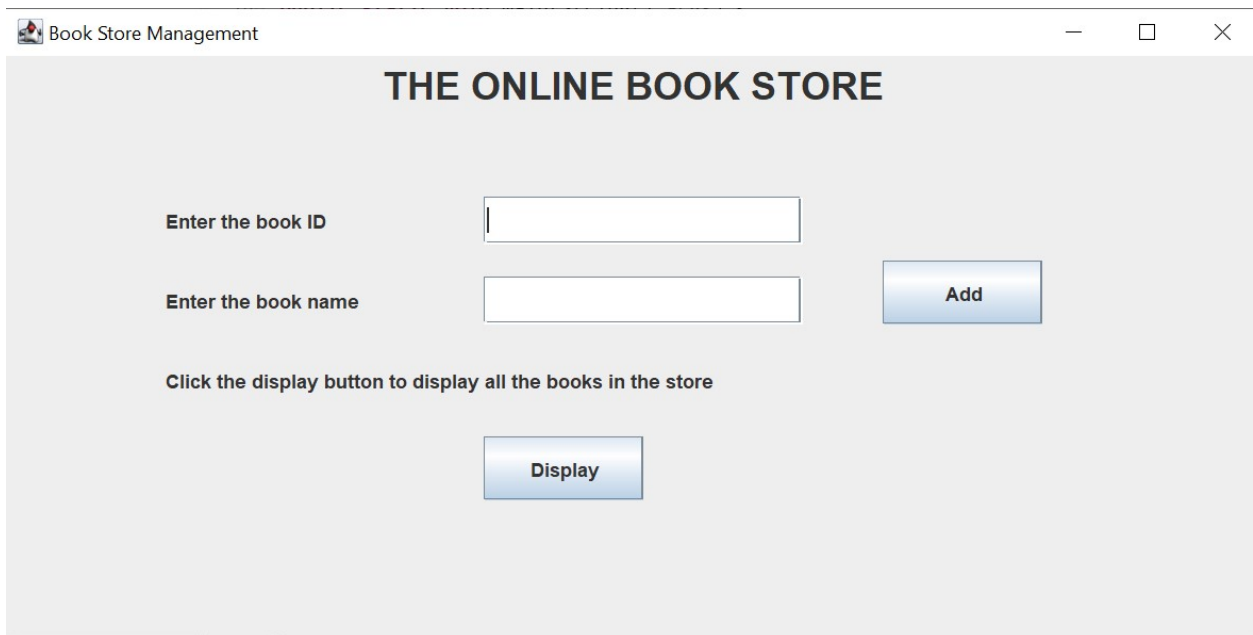**NAME : YUVARAJ V C**

REG NO: 727822TUCS261

DEPT: CSE

SECTION :C

SUBJECT: ADVANCED JAVA PROGRAMMING

CODING:

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.HashSet;
import java.util.Set;
public class book {
```

```java
private static Set<String> uniqueBookIds = new HashSet<>();
public static void main(String[] args) {
SwingUtilities.invokeLater(() -> createAndShowGUI());
}
private static void createAndShowGUI() {
JFrame frame = new JFrame("Book Store Management");
frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

JPanel backgroundPanel = new JPanel() {
@Override
protected void paintComponent(Graphics g) {
super.paintComponent(g);

}
};
backgroundPanel.setBackground(new Color(231, 231, 231)); // Light Blue color
backgroundPanel.setLayout(new BorderLayout());

JPanel headerPanel = new JPanel();
JLabel welcomeLabel = new JLabel("THE ONLINE BOOK STORE");
welcomeLabel.setFont(new Font("Arial", Font.BOLD, 24));
headerPanel.add(welcomeLabel);
backgroundPanel.add(headerPanel, BorderLayout.NORTH);
// Create a center panel with form components
JPanel centerPanel = new JPanel();
centerPanel.setLayout(null);
JLabel label4 = new JLabel("Enter the book ID");
JLabel label2 = new JLabel("Enter the book name ");
JLabel label3 = new JLabel("Click the display button to display all the books in the
store");
JTextField text1 = new JTextField();
JTextField text2 = new JTextField();
JButton button1 = new JButton("Add");
JButton button2 = new JButton("Display");

label4.setBounds(100, 50, 200, 30);
label2.setBounds(100, 100, 300, 30);
```

```java
label3.setBounds(100, 150, 400, 30);
text1.setBounds(300, 50, 200, 30);
text2.setBounds(300, 100, 200, 30);
button1.setBounds(550, 90, 100, 40);
button2.setBounds(300, 200, 100, 40);

centerPanel.add(label4);
centerPanel.add(label2);
centerPanel.add(label3);
centerPanel.add(text1);
centerPanel.add(text2);
centerPanel.add(button1);
centerPanel.add(button2);
backgroundPanel.add(centerPanel, BorderLayout.CENTER);

frame.setContentPane(backgroundPanel);
frame.setSize(800, 400);
frame.setLocationRelativeTo(null);
frame.setVisible(true);
// ActionListener for the Add button
button1.addActionListener(new ActionListener() {
@Override
public void actionPerformed(ActionEvent e) {
addBook(text1.getText(), text2.getText());
}
});

button2.addActionListener(new ActionListener() {
@Override
public void actionPerformed(ActionEvent e) {
displayBooks();
}
});
}
private static void addBook(String bookId, String bookName) {
try {
if (bookId.isEmpty() || bookName.isEmpty()) {
```

```java
JOptionPane.showMessageDialog(null, "Please enter both book ID and book  name.");
return;
}
if (uniqueBookIds.contains(bookId)) {
JOptionPane.showMessageDialog(null, "Book ID already exists.");
return;
}
Class.forName("com.mysql.cj.jdbc.Driver");
Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/store",
"root", "2647");
String sql = "INSERT INTO books (book_id, book_name) VALUES (?, ?)";
try (PreparedStatement ps = con.prepareStatement(sql)) {
ps.setString(1, bookId);
ps.setString(2, bookName);
ps.executeUpdate();
JOptionPane.showMessageDialog(null, "Book has been added successfully");
uniqueBookIds.add(bookId);
}
} catch (ClassNotFoundException | SQLException ex) {
ex.printStackTrace();
JOptionPane.showMessageDialog(null, "Error adding book: " + ex.getMessage());
}
}
private static void displayBooks() {
try {
Class.forName("com.mysql.cj.jdbc.Driver");
Connection con = DriverManager.getConnection("jdbc:mysql://localhost:3306/store",
"root", "2647");
String sql = "SELECT * FROM books";
try (PreparedStatement ps = con.prepareStatement(sql)) {
ResultSet rs = ps.executeQuery();
StringBuilder result = new StringBuilder("Book ID\tBook Name\n");
while (rs.next()) {

result.append(rs.getString("book_id")).append("\t").append(rs.getString("book_name")).append("\n");
}
```

```
 JOptionPane.showMessageDialog(null, result.toString(), "Book Store Inventory",
JOptionPane.INFORMATION_MESSAGE);
 }
 } catch (ClassNotFoundException | SQLException ex) {
 ex.printStackTrace();
 JOptionPane.showMessageDialog(null, "Error displaying books: " + ex.getMessage());
 }
 }


}
```

OUTPUT:

Book Store Inventory   ✕

(i)   **Book IDBook Name**
       1The lords of rings
       2Invisible MAN
       3ALICE adventures in wonderland
       4WHITE TIGER
       5SACRED GAMES
       6DISCOVERY OF INDIA

                    OK