

```
package project;
```

```
import java.time.Year;
```

```
import javax.swing.*;
```

```
import java.awt.*;
```

```
import java.awt.event.*;
```

```
import java.sql.*;
```

```
import java.util.logging.Level;
```

```
import java.util.logging.Logger;
```

```
class Login extends JFrame implements ActionListener {
```

```
    JTextField tf1 = new JTextField();
```

```
    JTextField tf2 = new JTextField();
```

```
    JTextField tf3 = new JTextField();
```

```
    JTextField tf4 = new JTextField();
```

```
    JTextField tf5 = new JTextField();
```

```
    JTextField tf6 = new JTextField();
```

```
    JTextField tf7 = new JTextField();
```

```
    JTextField tf8 = new JTextField();
```

```
    public Login() {
```

```
        Toolkit toolkit = Toolkit.getDefaultToolkit();
```

```
        Dimension screenSize = toolkit.getScreenSize();
```

```
        int screenWidth = screenSize.width;
```

```
        int screenHeight = screenSize.height;
```

```
        String imagePath = "C:/Program Files (x86)/NetBeans 8.2/flightBG.png";
```

```
        ImageIcon backgroundImage = new ImageIcon(imagePath);
```

```
        Image img = backgroundImage.getImage();
```

```
        Image scaledImg = img.getScaledInstance(screenWidth, screenHeight, Image.SCALE_SMOOTH);
```

```
        backgroundImage = new ImageIcon(scaledImg);
```

```

JLabel label = new JLabel(backgroundImage);
label.setBounds(0, 0, screenWidth, screenHeight);

JButton b1 = new JButton("SignIn");
b1.setBounds(850, 300, 150, 30);

JButton b2 = new JButton("SignUp");
b2.setBounds(850, 350, 150, 30);

JButton b3 = new JButton("Cancel");
b3.setBounds(850, 400, 150, 30);

label.add(b1);
label.add(b2);
label.add(b3);

b1.addActionListener(this);
b2.addActionListener(this);
b3.addActionListener(this);

add(label);
setSize(screenWidth, screenHeight); // Adjusted to fit the background
setVisible(true);
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
}

@Override
public void actionPerformed(ActionEvent e) {
    try {
        Class.forName("org.apache.derby.jdbc.ClientDriver");
    }
}

```

```
Connection con = DriverManager.getConnection("jdbc:derby://localhost:1527/FlightBooking",
"rohith", "rohith2002");
```

```
if (e.getActionCommand().equals("SignIn")) {
    openNewPage();
}
```

```
if (e.getActionCommand().equals("SignUp")) {
    openNewPage1();
}
```

```
if (e.getActionCommand().equals("Cancel")) {
    dispose();
}
```

```
con.close();
} catch (ClassNotFoundException ex) {
    Logger.getLogger(Login.class.getName()).log(Level.SEVERE, null, ex);
} catch (SQLException ex) {
    Logger.getLogger(Login.class.getName()).log(Level.SEVERE, null, ex);
}
}
```

```
public void openNewPage() {
    JFrame newPage = new JFrame("Old User Login");
    newPage.setSize(600, 600);
    newPage.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);

    String imagePath = "C:/Program Files (x86)/NetBeans 8.2/new.png";
    ImageIcon backgroundImage = new ImageIcon(imagePath);
    Image img = backgroundImage.getImage();
}
```

```
Image scaledImg = img.getScaledInstance(600, 600, Image.SCALE_SMOOTH);  
backgroundImage = new ImageIcon(scaledImg);  
JLabel label = new JLabel(backgroundImage);
```

```
label.setBounds(0, 0, 600, 600);  
newPage.add(label);
```

```
JLabel l1 = new JLabel("UserName: ");  
l1.setBounds(150, 200, 200, 30);  
l1.setFont(new Font("Arial", Font.PLAIN, 18));  
tf7.setBounds(250, 200, 300, 30);
```

```
JLabel l2 = new JLabel("Password: ");  
l2.setBounds(150, 250, 200, 30);  
l2.setFont(new Font("Arial", Font.PLAIN, 18));  
tf8.setBounds(250, 250, 300, 30);
```

```
JButton b1 = new JButton("SignIn");  
b1.setBounds(320, 400, 120, 30);  
JButton b2 = new JButton("Cancel");  
b2.setBounds(320, 450, 120, 30);
```

```
label.add(l1);  
label.add(l2);
```

```
label.add(tf7);  
label.add(tf8);  
label.add(b1);  
label.add(b2);
```

```
b1.addActionListener(new ActionListener() {
```

```
public void actionPerformed(ActionEvent e) {  
    signInUser();  
}  
});
```

```
b2.addActionListener(new ActionListener() {  
    public void actionPerformed(ActionEvent e) {  
        newPage.dispose();  
    }  
});
```

```
newPage.setVisible(true);  
}
```

```
public void signInUser() {  
    String s1 = tf7.getText();  
    String s2 = tf8.getText();
```

```
    if (s1.isEmpty() || s2.isEmpty()) {  
        JOptionPane.showMessageDialog(this, "All fields must be filled!");  
        return;  
    }
```

```
    try {  
        Class.forName("org.apache.derby.jdbc.ClientDriver");
```

```
        Connection con = DriverManager.getConnection("jdbc:derby://localhost:1527/FlightBooking",  
"rohith", "rohith2002");
```

```
        String query = "SELECT * FROM LOGIN WHERE UserName=? AND Password=?";  
        PreparedStatement pst = con.prepareStatement(query);
```

```

        pst.setString(1, s1);
        pst.setString(2, s2);

        ResultSet rs = pst.executeQuery();

        if (rs.next()) {
            openNewPage2();
        } else {
            JOptionPane.showMessageDialog(this, "Invalid Username or Password!");
        }

        con.close();
    } catch (ClassNotFoundException ex) {
        Logger.getLogger(Login.class.getName()).log(Level.SEVERE, null, ex);
        JOptionPane.showMessageDialog(this, "Database driver not found.");
    } catch (SQLException ex) {
        Logger.getLogger(Login.class.getName()).log(Level.SEVERE, null, ex);
        JOptionPane.showMessageDialog(this, "Error occurred while accessing the database.");
    }
}

public void openNewPage2() {
    JFrame newPage = new JFrame("Ticket Booking");
    newPage.setSize(600, 600);
    newPage.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);

    String imagePath = "C:/Program Files (x86)/NetBeans 8.2/new.png";
    ImageIcon backgroundImage = new ImageIcon(imagePath);
    Image img = backgroundImage.getImage();
    Image scaledImg = img.getScaledInstance(600, 600, Image.SCALE_SMOOTH);

```

```
backgroundImage = new ImageIcon(scaledImg);
```

```
JLabel label = new JLabel(backgroundImage);
```

```
label.setBounds(0, 0, 600, 600);
```

```
newPage.add(label);
```

```
JLabel l1 = new JLabel("Name: ");
```

```
l1.setBounds(150, 50, 200, 30);
```

```
l1.setFont(new Font("Arial", Font.PLAIN, 18));
```

```
tf4.setBounds(250, 50, 300, 30);
```

```
JLabel l2 = new JLabel("Age: ");
```

```
l2.setBounds(150, 100, 200, 30);
```

```
l2.setFont(new Font("Arial", Font.PLAIN, 18));
```

```
tf5.setBounds(250, 100, 300, 30);
```

```
JLabel l3 = new JLabel("Gender: ");
```

```
l3.setBounds(150, 150, 200, 30);
```

```
l3.setFont(new Font("Arial", Font.PLAIN, 18));
```

```
JRadioButton maleButton = new JRadioButton("Male");
```

```
maleButton.setBounds(250, 150, 70, 30);
```

```
JRadioButton femaleButton = new JRadioButton("Female");
```

```
femaleButton.setBounds(320, 150, 80, 30);
```

```
ButtonGroup genderGroup = new ButtonGroup();
```

```
genderGroup.add(maleButton);
```

```
genderGroup.add(femaleButton);
```

```
label.add(l1);
```

```
label.add(l2);
```

```
label.add(l3);  
label.add(tf4);  
label.add(tf5);  
label.add(maleButton);  
label.add(femaleButton);
```

```
JLabel l4 = new JLabel("From: ");  
l4.setBounds(150, 200, 200, 30);  
l4.setFont(new Font("Arial", Font.PLAIN, 18));  
JTextField fromField = new JTextField();  
fromField.setBounds(250, 200, 300, 30);
```

```
JLabel l5 = new JLabel("To: ");  
l5.setBounds(150, 250, 200, 30);  
l5.setFont(new Font("Arial", Font.PLAIN, 18));  
JTextField toField = new JTextField();  
toField.setBounds(250, 250, 300, 30);
```

```
label.add(l4);  
label.add(fromField);  
label.add(l5);  
label.add(toField);
```

```
JLabel l6 = new JLabel("Travel date: ");  
l6.setBounds(150, 300, 200, 30);  
l6.setFont(new Font("Arial", Font.PLAIN, 18));  
label.add(l6);
```

```
String[] days = new String[31];  
for (int i = 1; i <= 31; i++) {  
    days[i - 1] = String.valueOf(i);
```



```
}
```

```
String[] months = new String[12];  
for (int i = 1; i <= 12; i++) {  
    months[i - 1] = String.valueOf(i);  
}
```

```
String[] years = new String[100];  
for (int i = 0; i < 100; i++) {  
    years[i] = String.valueOf(Year.now().getValue() + i);  
}
```

```
JComboBox<String> dayComboBox = new JComboBox<>(days);  
JComboBox<String> monthComboBox = new JComboBox<>(months);  
JComboBox<String> yearComboBox = new JComboBox<>(years);
```

```
dayComboBox.setBounds(250, 300, 50, 30);  
monthComboBox.setBounds(310, 300, 50, 30);  
yearComboBox.setBounds(370, 300, 70, 30);
```

```
label.add(dayComboBox);  
label.add(monthComboBox);  
label.add(yearComboBox);
```

```
JButton b1 = new JButton("Book Ticket");  
b1.setBounds(200, 400, 120, 30);  
JButton b2 = new JButton("Cancel");  
b2.setBounds(350, 400, 120, 30);
```

```
label.add(b1);  
label.add(b2);
```

```

b1.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        String name = tf4.getText();
        String age = tf5.getText();
        String from = fromField.getText();
        String to = toField.getText();

        String travelDate = dayComboBox.getSelectedItem() + "/" +
monthComboBox.getSelectedItem() + "/" + yearComboBox.getSelectedItem();

        if (name.isEmpty() || age.isEmpty() || from.isEmpty() || to.isEmpty()) {
            JOptionPane.showMessageDialog(newPage, "All fields must be filled!");
            return;
        }

        // Simulate booking logic here, such as storing in a database or processing

        JOptionPane.showMessageDialog(newPage, "Ticket booked successfully for " + name +
            "! From: " + from + " To: " + to + " on " + travelDate);
        newPage.dispose(); // Close the booking page after successful booking
    }
});

b2.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        newPage.dispose();
    }
});

newPage.setVisible(true);
}

```

```

public void bookTicket() {
    String name = tf4.getText();
    String age = tf5.getText();

    if (name.isEmpty() || age.isEmpty()) {
        JOptionPane.showMessageDialog(this, "All fields must be filled!");
        return;
    }

    // Additional logic for booking can be implemented here

    JOptionPane.showMessageDialog(this, "Ticket booked successfully!");
}

public void openNewPage1() {
    JFrame newPage = new JFrame("New User Registration");
    newPage.setSize(600, 600);
    newPage.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);

    String imagePath = "C:/Program Files (x86)/NetBeans 8.2/new.png";
    ImageIcon backgroundImage = new ImageIcon(imagePath);
    Image img = backgroundImage.getImage();
    Image scaledImg = img.getScaledInstance(600, 600, Image.SCALE_SMOOTH);
    backgroundImage = new ImageIcon(scaledImg);
    JLabel label = new JLabel(backgroundImage);

    label.setBounds(0, 0, 600, 600);
    newPage.add(label);

    JLabel l1 = new JLabel("UserName: ");

```

```
l1.setBounds(150, 200, 200, 30);  
l1.setFont(new Font("Arial", Font.PLAIN, 18));  
tf1.setBounds(250, 200, 300, 30);
```

```
JLabel l2 = new JLabel("Password: ");  
l2.setBounds(150, 250, 200, 30);  
l2.setFont(new Font("Arial", Font.PLAIN, 18));  
tf2.setBounds(250, 250, 300, 30);
```

```
JLabel l3 = new JLabel("Email: ");  
l3.setBounds(150, 300, 200, 30);  
l3.setFont(new Font("Arial", Font.PLAIN, 18));  
tf3.setBounds(250, 300, 300, 30);
```

```
JLabel l4 = new JLabel("Phone: ");  
l4.setBounds(150, 350, 200, 30);  
l4.setFont(new Font("Arial", Font.PLAIN, 18));  
tf4.setBounds(250, 350, 300, 30);
```

```
label.add(l1);  
label.add(l2);  
label.add(l3);  
label.add(l4);
```

```
label.add(tf1);  
label.add(tf2);  
label.add(tf3);  
label.add(tf4);
```

```
JButton b1 = new JButton("SignUp");  
b1.setBounds(200, 450, 120, 30);
```

```

JButton b2 = new JButton("Cancel");

b2.setBounds(350, 450, 120, 30);

label.add(b1);
label.add(b2);

b1.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        signUpUser();
    }
});

b2.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent e) {
        newPage.dispose();
    }
});

newPage.setVisible(true);
}

public void signUpUser() {
    String username = tf1.getText();
    String password = tf2.getText();
    String email = tf3.getText();
    String phone = tf4.getText();

    if (username.isEmpty() || password.isEmpty() || email.isEmpty() || phone.isEmpty()) {
        JOptionPane.showMessageDialog(this, "All fields must be filled!");
        return;
    }
}

```

```

try {
    Class.forName("org.apache.derby.jdbc.ClientDriver");

    Connection con = DriverManager.getConnection("jdbc:derby://localhost:1527/FlightBooking",
"rohith", "rohith2002");

    String query = "INSERT INTO LOGIN (UserName, Password, Email, Phone) VALUES (?, ?, ?, ?)";
    PreparedStatement pst = con.prepareStatement(query);
    pst.setString(1, username);
    pst.setString(2, password);
    pst.setString(3, email);
    pst.setString(4, phone);

    int rowsAffected = pst.executeUpdate();
    if (rowsAffected > 0) {
        JOptionPane.showMessageDialog(this, "Registration successful!");
    } else {
        JOptionPane.showMessageDialog(this, "Registration failed!");
    }

    con.close();
} catch (ClassNotFoundException ex) {
    Logger.getLogger(Login.class.getName()).log(Level.SEVERE, null, ex);
    JOptionPane.showMessageDialog(this, "Database driver not found.");
} catch (SQLException ex) {
    Logger.getLogger(Login.class.getName()).log(Level.SEVERE, null, ex);
    JOptionPane.showMessageDialog(this, "Error occurred while accessing the database.");
}
}

public static void main(String[] args) {

```

```
        new Login();  
    }  
}
```