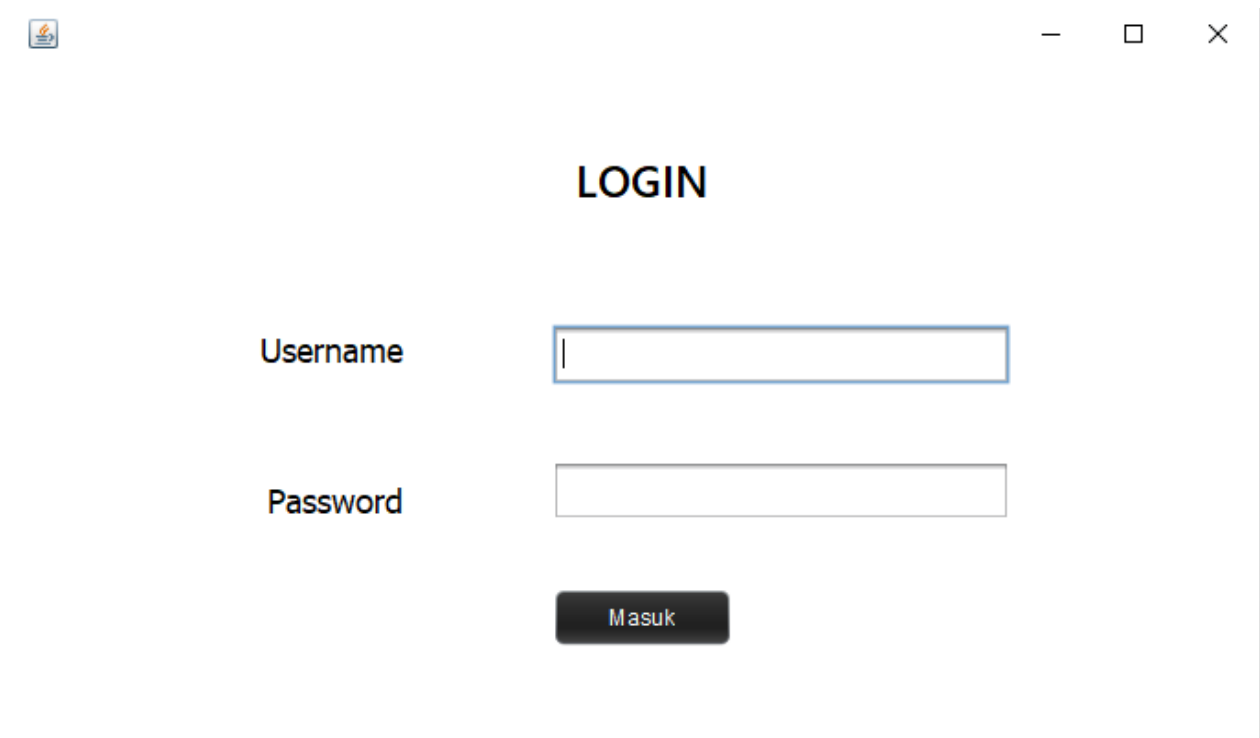


Nama : Rian Firgiawan Gusti Ananda

NIM : 3130021008

### Halaman Login



LOGIN

Username

Password

Masuk

### Source Code :

```
import java.sql.Connection;  
import java.sql.ResultSet;  
import java.sql.SQLException;  
import java.sql.Statement;  
import javax.swing.JOptionPane;
```

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    String user, pass;
    user = username.getText();
    pass = password.getText();

    try {
        String sql = "SELECT * FROM users WHERE username='" + user + "' && password='" + pass + "'";
        Connection con = Koneksi.getKoneksi();
        Statement st = con.createStatement();

        ResultSet rs = st.executeQuery(sql);

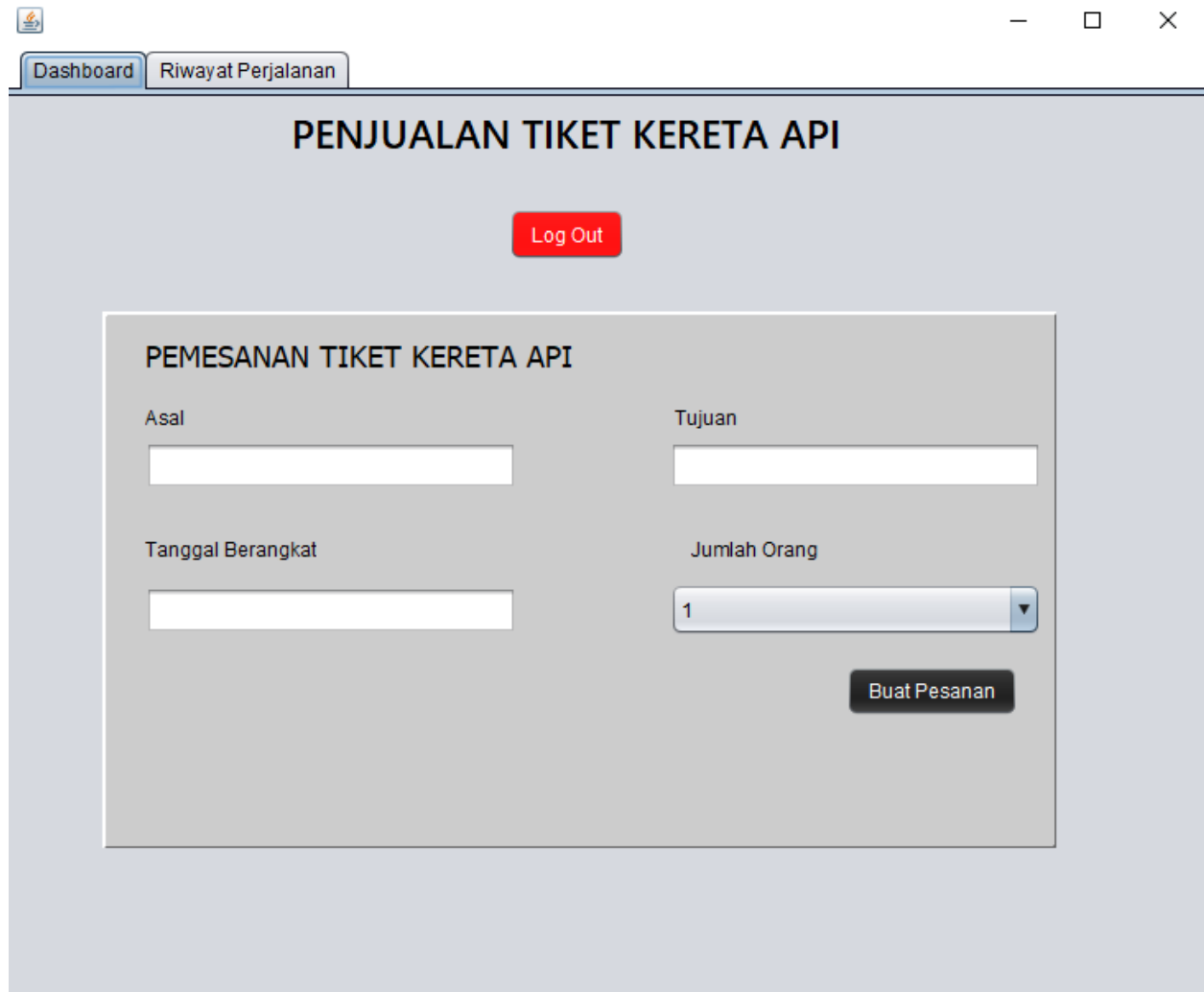
        if(rs.next()){
            Statement st2 = con.createStatement();
            int rs2 = st2.executeUpdate("INSERT INTO login(user) VALUES('" + user + "')");
            if(rs2 == 0){
                JOptionPane.showMessageDialog(null, "Anda Gagal Login!");
            }else {
                JOptionPane.showMessageDialog(null, "Anda Berhasil Login!");
                Dashboard home = new Dashboard();
                home.setVisible(true);
                dispose();
            }
        }else{
            JOptionPane.showMessageDialog(null, "Anda Gagal Login!");
        }

    } catch (SQLException ex) {

    }
}

```

## Halaman Dashboard



The screenshot shows a web browser window with a dashboard titled "PENJUALAN TIKET KERETA API". At the top left, there are two tabs: "Dashboard" (active) and "Riwayat Perjalanan". At the top right, there are window control buttons (minimize, maximize, close). Below the title, there is a red "Log Out" button. The main content area contains a form titled "PEMESANAN TIKET KERETA API". The form has four input fields: "Asal" (origin), "Tujuan" (destination), "Tanggal Berangkat" (departure date), and "Jumlah Orang" (number of people). The "Jumlah Orang" field is a dropdown menu currently set to "1". A dark grey "Buat Pesanan" (Make Order) button is located at the bottom right of the form.

Dashboard Riwayat Perjalanan

### PENJUALAN TIKET KERETA API

Log Out

#### PEMESANAN TIKET KERETA API

Asal

Tujuan

Tanggal Berangkat

Jumlah Orang

Buat Pesanan

Source Code :

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

    // TODO add your handling code here:

    Connection con = Koneksi.getKoneksi();
    Statement st, st1;
    String input1, input2, input3, input4, input5, input6, input7, input8, input9, user = "";
    int id = 0, kode = 0, jumlah = 0, harga = 0, total;
    Random rand = new Random();
    kode = rand.nextInt(50) * 12345;
    try {
        st = con.createStatement();
        ResultSet rs = st.executeQuery("SELECT user FROM login");
        if (rs.next()) {
            user = rs.getString("user");
        }

        input1 = asal.getText();
        input2 = tujuan.getText();
        input3 = tgl.getText();
        st.executeUpdate("INSERT INTO riwayat(asal,tujuan,tgl,user) VALUES ('" + input1 + "','" + input
2 + "','" + input3 + "','" + user + "')");

        ResultSet result = st.executeQuery("SELECT * FROM tagihan WHERE kode=" + kode);
        if (result.next()) {
            kode = rand.nextInt(50) * 12345;
        }
        int tagihan = st.executeUpdate("INSERT INTO tagihan(kode) VALUES (" + kode + ")");

        if (penumpang1.getText().length() == 0) {
            input4 = null;
            input7 = null;
        } else {

```

```

        input4 = penumpang1.getText();
        input7 = (String) pilihan1.getSelectedItem();
        if (input7.contains("Ekonomi")) {
            harga = 20000;
        } else if (input7.contains("Eksekutif")) {
            harga = 30000;
        } else if (input7.contains("Bisnis")) {
            harga = 50000;
        }

        st1 = con.createStatement();
        ResultSet result2 = st1.executeQuery("SELECT * FROM tagihan ORDER BY id DESC");
        if (result2.next()) {
            id = result2.getInt("id");
        }

        st.executeUpdate("INSERT INTO pesanan(tagihan_id,nama,asal,tujuan,tgl,jenis_kereta,harga) V
VALUES ('" + id + "','" + input4 + "','" + input1 + "','" + input2 + "','" + input3 + "','" + input7 + "','"
+ harga + "')");
    }

    if (penumpang2.getText().length() == 0) {
        input5 = null;
        input8 = null;
    } else {
        input5 = penumpang2.getText();
        input8 = (String) pilihan2.getSelectedItem();
        if (input8.contains("Ekonomi")) {
            harga = 20000;
        } else if (input8.contains("Eksekutif")) {
            harga = 30000;
        } else if (input8.contains("Bisnis")) {
            harga = 50000;
        }
    }

```

```

        st1 = con.createStatement();
        ResultSet result2 = st1.executeQuery("SELECT * FROM tagihan ORDER BY id DESC");
        if (result2.next()) {
            id = result2.getInt("id");
        }
        st.executeUpdate("INSERT INTO pesanan(tagihan_id,nama,asal,tujuan,tgl,jenis_kereta,harga) V
VALUES ('" + id + "','" + input5 + "','" + input1 + "','" + input2 + "','" + input3 + "','" + input8 + "','"
+ harga + "')");
    }
    if (penumpang3.getText().length() == 0) {
        input6 = null;
        input9 = null;
    } else {
        input6 = penumpang3.getText();
        input9 = (String) pilihan3.getSelectedItem();
        if (input8.contains("Ekonomi")) {
            harga = 20000;
        } else if (input8.contains("Eksekutif")) {
            harga = 30000;
        } else if (input8.contains("Bisnis")) {
            harga = 50000;
        }

        st1 = con.createStatement();
        ResultSet result2 = st1.executeQuery("SELECT * FROM tagihan ORDER BY id DESC");
        if (result2.next()) {
            id = result2.getInt("id");
        }
        st.executeUpdate("INSERT INTO pesanan(tagihan_id,nama,asal,tujuan,tgl,jenis_kereta,harga) V
VALUES ('" + id + "','" + input6 + "','" + input1 + "','" + input2 + "','" + input3 + "','" + input9 + "','"
+ harga + "')");
    }
}

```

```

        Tagihan tg = new Tagihan();
        tg.setVisible(true);
        dispose();

    } catch (SQLException ex) {
        Logger.getLogger(Dashboard.class.getName()).log(Level.SEVERE, null, ex);
    }
}

```

```

private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {

    // TODO add your handling code here:
    Connection con = Koneksi.getKoneksi();
    try {
        Statement st = con.createStatement();
        int result = st.executeUpdate("DELETE FROM login");
        if (result == 0) {
            JOptionPane.showMessageDialog(null, "Anda Gagal Logout!");
        } else {
            Login lg = new Login();
            lg.setVisible(true);
            dispose();
        }
    } catch (SQLException ex) {
        Logger.getLogger(Dashboard.class.getName()).log(Level.SEVERE, null, ex);
    }
}

```

```
private void buatActionPerformed(java.awt.event.ActionEvent evt) {
```

```
    // TODO add your handling code here:
```

```
    labelDaftar.setVisible(true);
```

```
    jButton1.setVisible(true);
```

```
    if (jumlah.getSelectedItemAt() == "1") {
```

```
        penumpang1.setVisible(true);
```

```
        label11.setVisible(true);
```

```
        label14.setVisible(true);
```

```
        pilihan1.setVisible(true);
```

```
        // penumpang 2
```

```
        penumpang2.setVisible(false);
```

```
        label12.setVisible(false);
```

```
        label15.setVisible(false);
```

```
        pilihan2.setVisible(false);
```

```
        // penumpang 3
```

```
        penumpang3.setVisible(false);
```

```
        label13.setVisible(false);
```

```
        label16.setVisible(false);
```

```
        pilihan3.setVisible(false);
```

```
    } else if (jumlah.getSelectedItemAt() == "2") {
```

```
        // penumpang 1
```

```
        penumpang1.setVisible(true);
```

```
        label11.setVisible(true);
```

```
        label14.setVisible(true);
```

```
        pilihan1.setVisible(true);
```

```
        // penumpang 2
```

```
        penumpang2.setVisible(true);
```

```
        label12.setVisible(true);
```

```
        label15.setVisible(true);
```

```
        pilihan2.setVisible(true);
```

```
        // penumpang 3
```

```
        penumpang3.setVisible(false);
```

```
        label13.setVisible(false);
```

```
        label16.setVisible(false);
```

```
        pilihan3.setVisible(false);
```

```
    } else if (jumlah.getSelectedItemAt() == "3") {
```

```
        // penumpang 1
```

```
        penumpang1.setVisible(true);
```

```
        label11.setVisible(true);
```

```
        label14.setVisible(true);
```

```
        pilihan1.setVisible(true);
```

```
        // penumpang 2
```

```
        penumpang2.setVisible(true);
```

```
        label12.setVisible(true);
```

```
        label15.setVisible(true);
```

```
        pilihan2.setVisible(true);
```

```
        // penumpang 3
```

```
        penumpang3.setVisible(true);
```

```
        label13.setVisible(true);
```

```
        label16.setVisible(true);
```

```
        pilihan3.setVisible(true);
```

```
    }
```

```
}
```

RIWAYAT PERJALANAN		
Asal Kota	Tujuan Kota	Tanggal
Malang	Jakarta	23 Januari 2023
Surabaya	Sidoarjo	24 Januari 2023
Surabaya	Malang	19 Februari 2023
Surabaya	Malang	24 Januari 2023
Surabaya	Yogyakarta	24 Januari 2023
Sidoarjo	Jakarta	24 Januari 2023

## Source Code :

```
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;
import java.util.Random;
```



```

public Dashboard() {
    initComponents();

    penumpang1.setVisible(false);
    penumpang2.setVisible(false);
    penumpang3.setVisible(false);
    pilihan1.setVisible(false);
    pilihan2.setVisible(false);
    pilihan3.setVisible(false);
    label1.setVisible(false);
    label2.setVisible(false);
    label3.setVisible(false);
    label4.setVisible(false);
    label5.setVisible(false);
    label6.setVisible(false);
    jButton1.setVisible(false);
    labelDaftar.setVisible(false);
    tabel();
}

```

```

private void tabel() {
    Object[] row = {"Surabaya", "Semarang", "27 Juli 2023", "2", "Bisnis"};
    DefaultTableModel model = (DefaultTableModel) jTable1.getModel();
    model.setRowCount(0);
    try {
        Connection con = Koneksi.getKoneksi();
        Statement st = con.createStatement();
        ResultSet rs = st.executeQuery("SELECT user FROM login");
        if (rs.next()) {
            ResultSet result = st.executeQuery("SELECT * FROM riwayat WHERE user = '" + rs.getString("u
ser") + "'");
            while (result.next()) {
                model.addRow(new Object[]{
                    result.getString("asal"),
                    result.getString("tujuan"),
                    result.getString("tgl")});
            }
        }
    } catch (SQLException ex) {
        Logger.getLogger(Dashboard.class.getName()).log(Level.SEVERE, null, ex);
    }
}

```

## Halaman Tagihan



The screenshot shows a Java Swing window with a light gray background. At the top, there is a title bar with a small icon on the left and standard window controls (minimize, maximize, close) on the right. The main content area has the title "DAFTAR TAGIHAN" in bold black text. Below the title, there are four rows of text, each with a label on the left and a value on the right. The first row is "KODE PESANAN" with value "#543180". The second row is "NAMA PEMESAN" with value "rianfga". The third row is "TOTAL" with value "120000". The fourth row is "NOMINAL BAYAR" with a white rectangular text input field. At the bottom of the window, there are two dark gray buttons with white text: "Print" and "Bayar".

KODE PESANAN	#543180
NAMA PEMESAN	rianfga
TOTAL	120000
NOMINAL BAYAR	<input type="text"/>

Print Bayar

## Source Code :

```
import java.io.File;
import java.sql.Connection;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.HashMap;
import java.util.logging.Level;
import java.util.logging.Logger;
import javax.swing.JOptionPane;
import net.sf.jasperreports.engine.JRException;
import net.sf.jasperreports.engine.JasperFillManager;
import net.sf.jasperreports.engine.JasperPrint;
import net.sf.jasperreports.engine.JasperReport;
import net.sf.jasperreports.engine.util.JRLoader;
import net.sf.jasperreports.view.JasperViewer;
```

```

int id;
/**
 * Creates new form Tagihan
 */
public Tagihan() {
    initComponents();
    tampilkanData();
}

private void tampilkanData() {
    Connection con = Koneksi.getKoneksi();
    Statement st;
    String jenisK = "", user = "";
    int harga = 0, pembayaran = 0, jumlahOrang = 0;
    try {
        String sql = "SELECT kode,user,SUM(harga) AS total FROM pesanan,login,tagihan;";
        st = con.createStatement();
        ResultSet rs = st.executeQuery(sql);
        if (rs.next()) {
            id = rs.getInt("kode");
            user = rs.getString("user");
            pembayaran = rs.getInt("total");
        }

        kode.setText("#"+Integer.toString(id));
        nama.setText(user);
        total.setText(Integer.toString(pembayaran));
    } catch (SQLException ex) {
        Logger.getLogger(Tagihan.class.getName()).log(Level.SEVERE, null, ex);
    }
}

```

```

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    int nominalUser = Integer.parseInt(nominal.getText());
    int pembayaran = Integer.parseInt(total.getText());
    int bayar = nominalUser - pembayaran;
    JOptionPane.showMessageDialog(null, "Kembalian : " + nominalUser + " - " + pembayaran + " = " + bayar);

    String sql = "DELETE FROM pesanan";
    Connection con = Koneksi.getKoneksi();
    Statement st;
    try {
        st = con.createStatement();
        int rs = st.executeUpdate(sql);
        if (rs == 0) {
            JOptionPane.showMessageDialog(null, "Data Tidak Masuk!");
        } else {
            int rs2 = st.executeUpdate("DELETE FROM tagihan");
            if (rs2 == 0) {
                JOptionPane.showMessageDialog(null, "Data Tidak Masuk!");
            }
            Dashboard db = new Dashboard();
            db.setVisible(true);
            dispose();
        }
    } catch (SQLException ex) {
        Logger.getLogger(Tagihan.class.getName()).log(Level.SEVERE, null, ex);
    }
}

```

```

private void printActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    try {
        String NamaFile = "src/laporan/report.jasper";

        HashMap hash = new HashMap();

        File file = new File(NamaFile);
        JasperReport jasperReport = (JasperReport) JRLoader.loadObject(file.getPath());
        JasperPrint jasperPrint = JasperFillManager.fillReport(jasperReport, hash, Koneksi.getKoneksi());

        JasperViewer.viewReport(jasperPrint, false);
    } catch (JRException e) {
        System.out.println(e);
    }
}

```

Source Code Lainnya :

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

/**
 *
 * @author Rian
 */
public class Koneksi {
    // deklarasi variabel
    static Connection con;
    static String url;

    public static Connection getKoneksi() {
        try {
            // memanggil driver MySQL
            Class.forName("com.mysql.jdbc.Driver");

            // set url koneksi
            url = "jdbc:mysql://localhost:3306/app_pbouas";

            // membuat koneksi ke database
            con = DriverManager.getConnection(url, "root", "");
        } catch (ClassNotFoundException | SQLException e) {
            System.err.println(e);
        }

        return con;
    }
}
```

**Deskripsi Aplikasi :**

Aplikasi Penjualan Tiket Kereta Api ini adalah program penjualan tiket yang berbasis desktop. Program ini menggunakan java untuk bahasa pemrogramannya, program ini memiliki fitur antara lain : membuat pesanan tiket kereta, menampilkan invoice customer, membuat report invoice berupa pdf dan melihat daftar riwayat perjalanan. Dalam fitur ini juga terdapat plugin yang digunakan untuk membuat report berupa pdf yaitu ireport jasper.