SRM
UNIVERSITY AP
Andhra Pradesh

Branch of Computer Science and Engineering

Project

Employee Management Application

Submitted by:

Group 5

AP21110011491 - Dhanushree Y

AP21110011516 - Kalyan Ram P

AP21110011536 - Madhuri K

AP21110011548 - Hima Teja G

Course:

Java Programming

Guided by:

Mr. Sudhakar Reddy N

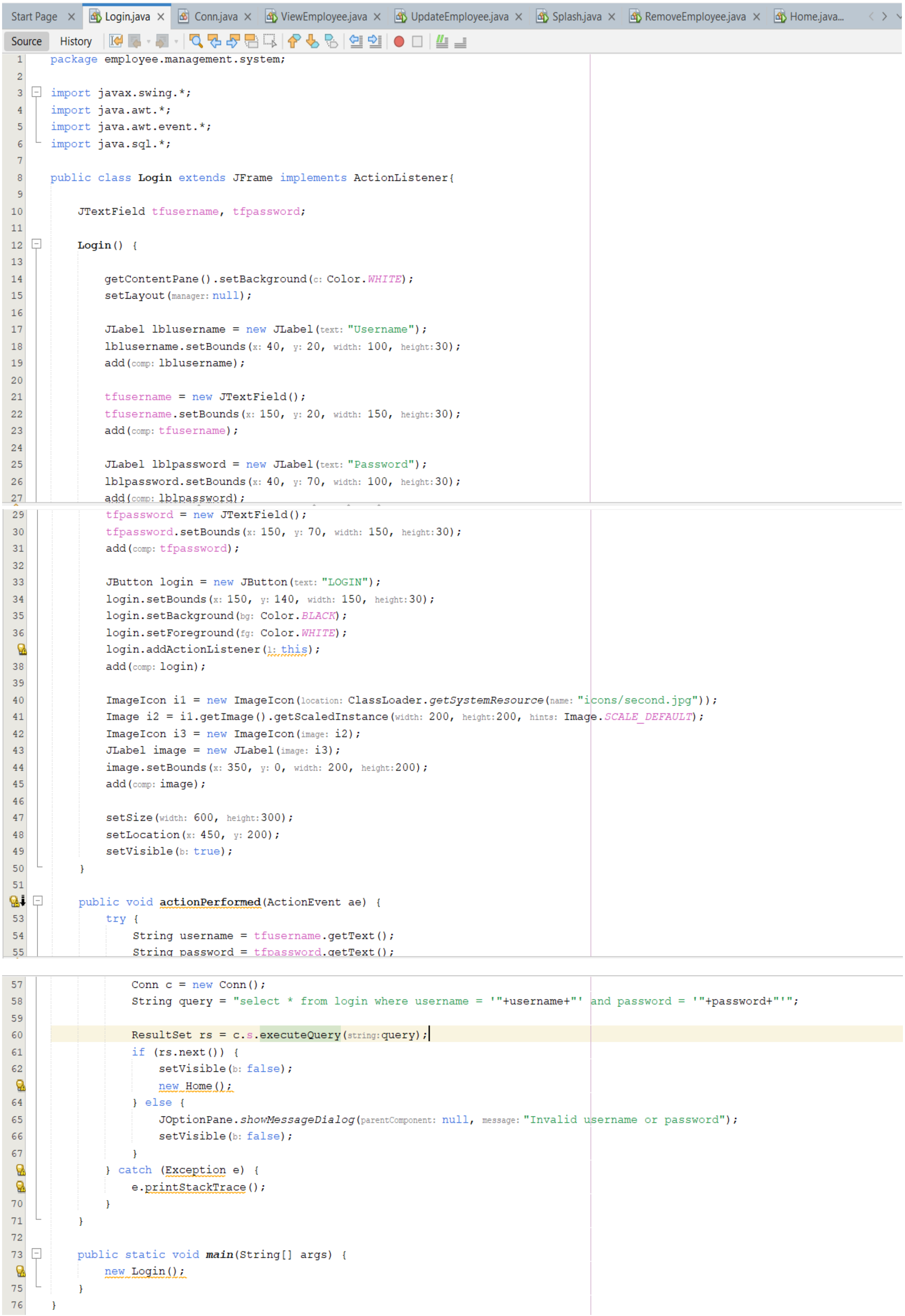
Contents

Abstract

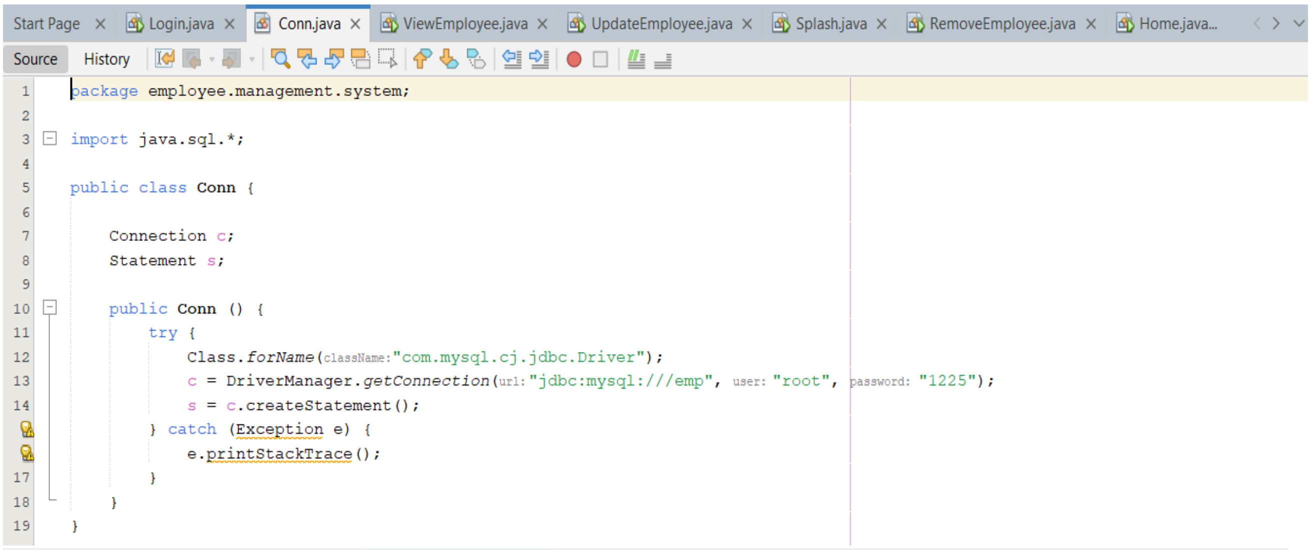
Employee Management Application helps us to maintain the details of employees working in an organization. Earlier people used records to maintain the details of an employee which was comparatively difficult to retrieve, access, or update the data of a person. Also, there was a lack of confidentiality in managing the details and the chances of losing the data were likely high.

Using this application, we can easily create, add, update, access, delete, and view an employee’s data as required. Also, it can be accessed only by authorized persons unlike when maintained in records as physical information. The chances of losing the data are insignificant. Thus, the data is confidential and secured.

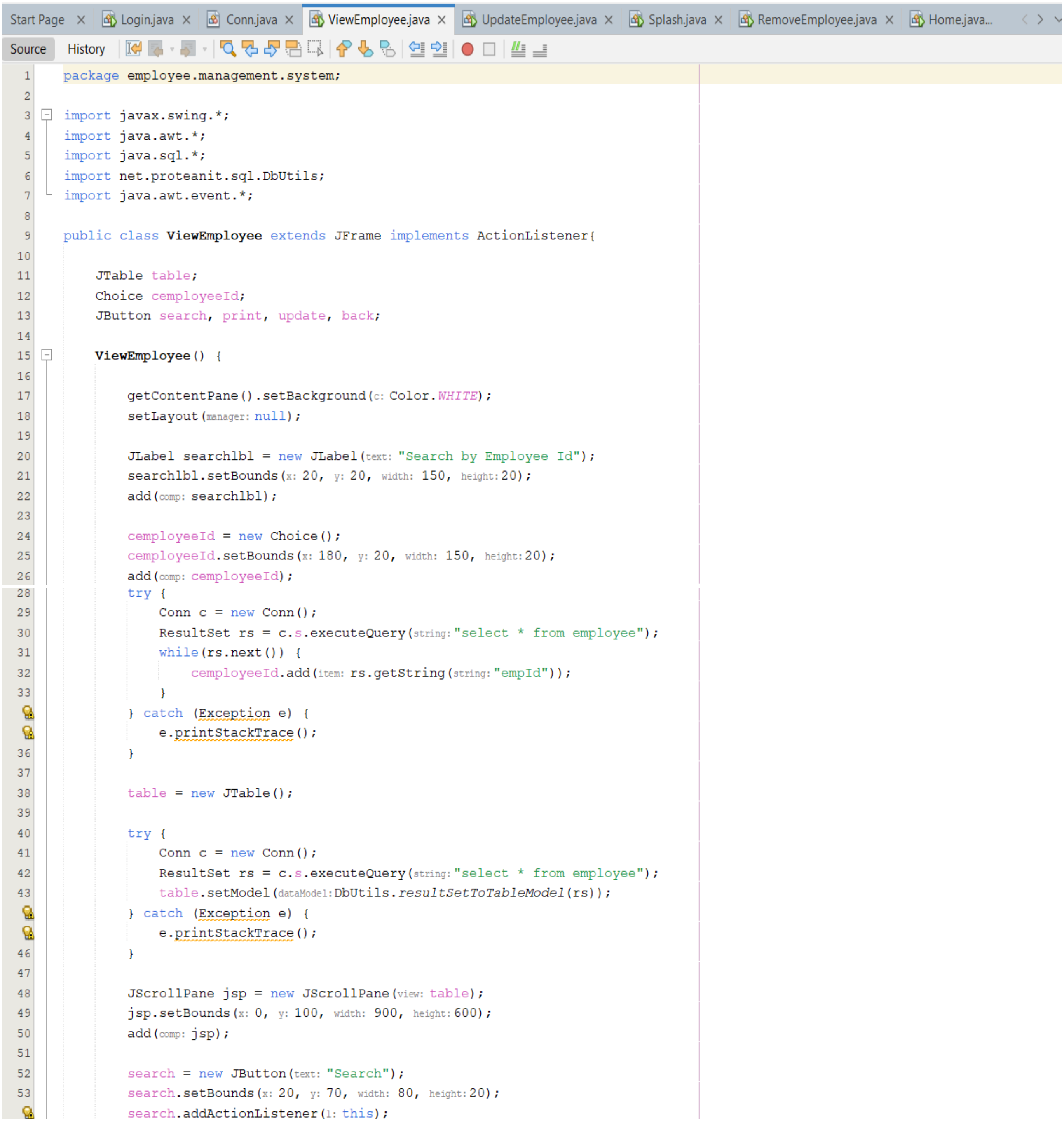
Source Code



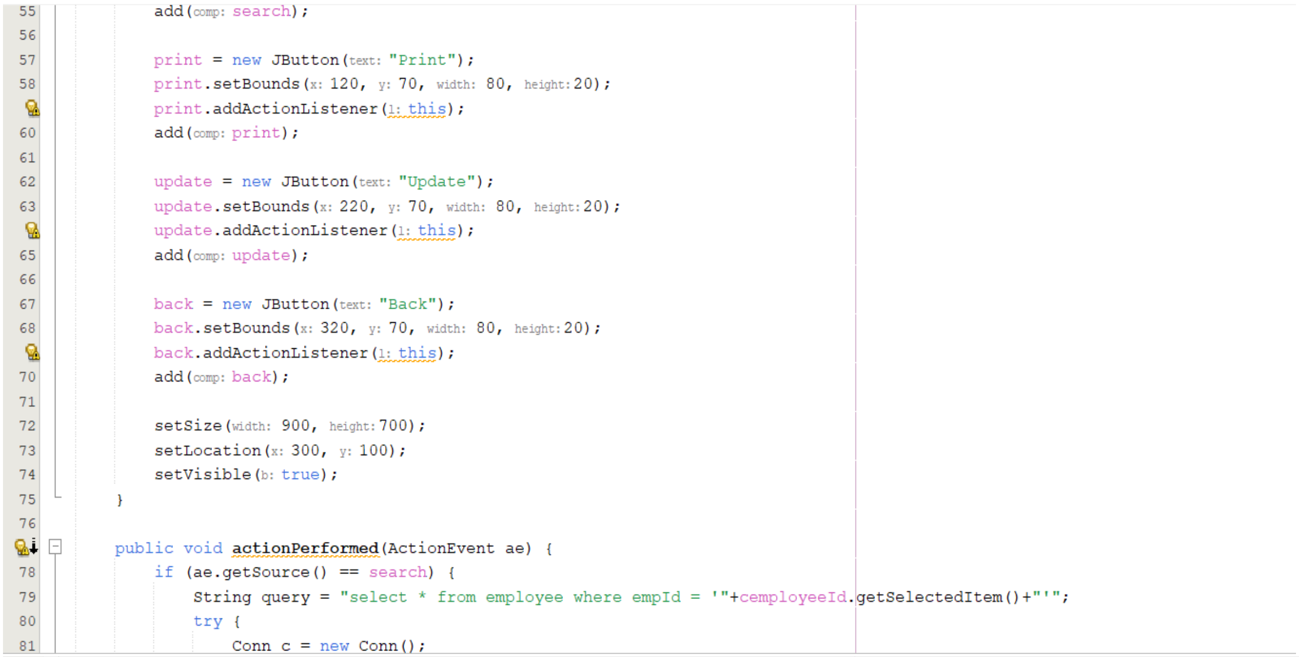
```java
package employee.management.system;

import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
import java.sql.*;

public class Login extends JFrame implements ActionListener{

    JTextField tfusername, tfpassword;

    Login() {

        getContentPane().setBackground(c: Color.WHITE);
        setLayout(manager: null);

        JLabel lblusername = new JLabel(text: "Username");
        lblusername.setBounds(x: 40, y: 20, width: 100, height:30);
        add(comp: lblusername);

        tfusername = new JTextField();
        tfusername.setBounds(x: 150, y: 20, width: 150, height:30);
        add(comp: tfusername);

        JLabel lblpassword = new JLabel(text: "Password");
        lblpassword.setBounds(x: 40, y: 70, width: 100, height:30);
        add(comp: lblpassword);
        tfpassword = new JTextField();
        tfpassword.setBounds(x: 150, y: 70, width: 150, height:30);
        add(comp: tfpassword);

        JButton login = new JButton(text: "LOGIN");
        login.setBounds(x: 150, y: 140, width: 150, height:30);
        login.setBackground(bg: Color.BLACK);
        login.setForeground(fg: Color.WHITE);
        login.addActionListener(l: this);
        add(comp: login);

        ImageIcon i1 = new ImageIcon(location: ClassLoader.getSystemResource(name: "icons/second.jpg"));
        Image i2 = i1.getImage().getScaledInstance(width: 200, height:200, hints: Image.SCALE_DEFAULT);
        ImageIcon i3 = new ImageIcon(image: i2);
        JLabel image = new JLabel(image: i3);
        image.setBounds(x: 350, y: 0, width: 200, height:200);
        add(comp: image);

        setSize(width: 600, height:300);
        setLocation(x: 450, y: 200);
        setVisible(b: true);
    }

    public void actionPerformed(ActionEvent ae) {
        try {
            String username = tfusername.getText();
            String password = tfpassword.getText();
            Conn c = new Conn();
            String query = "select * from login where username = '"+username+"' and password = '"+password+"'";

            ResultSet rs = c.s.executeQuery(string:query);
            if (rs.next()) {
                setVisible(b: false);
                new Home();
            } else {
                JOptionPane.showMessageDialog(parentComponent: null, message: "Invalid username or password");
                setVisible(b: false);
            }
        } catch (Exception e) {
            e.printStackTrace();
        }
    }

    public static void main(String[] args) {
        new Login();
    }
}
```



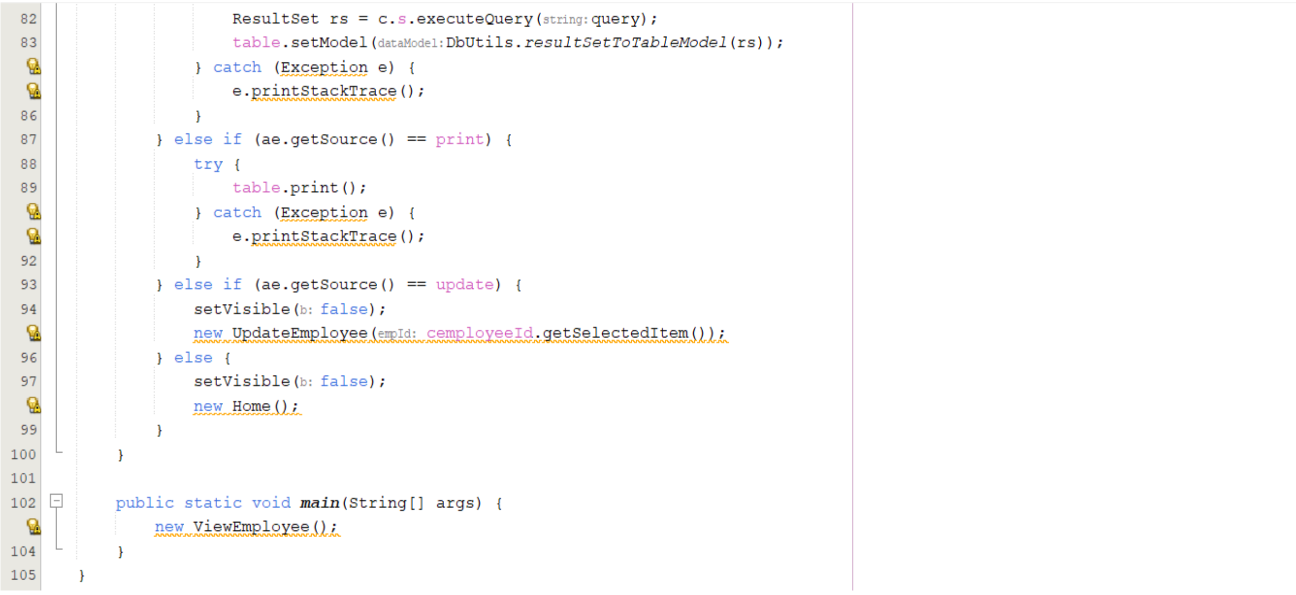
```
package employee.management.system;

import java.sql.*;

public class Conn {

    Connection c;
    Statement s;

    public Conn () {
        try {
            Class.forName(className:"com.mysql.cj.jdbc.Driver");
            c = DriverManager.getConnection(url:"jdbc:mysql:///emp", user: "root", password: "1225");
            s = c.createStatement();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```



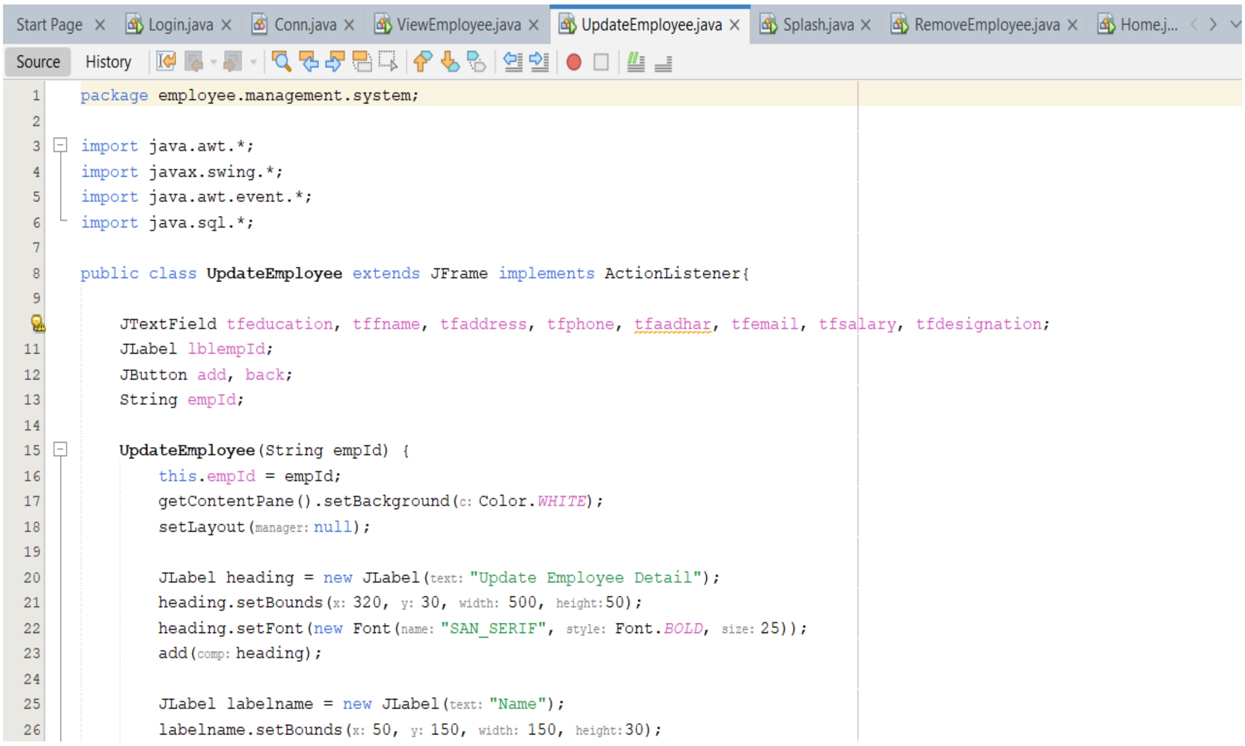
```
package employee.management.system;

import javax.swing.*;
import java.awt.*;
import java.sql.*;
import net.proteanit.sql.DbUtils;
import java.awt.event.*;

public class ViewEmployee extends JFrame implements ActionListener{

    JTable table;
    Choice cemployeeId;
    JButton search, print, update, back;

    ViewEmployee() {

        getContentPane().setBackground(c: Color.WHITE);
        setLayout(manager: null);

        JLabel searchlbl = new JLabel(text: "Search by Employee Id");
        searchlbl.setBounds(x: 20, y: 20, width: 150, height: 20);
        add(comp: searchlbl);

        cemployeeId = new Choice();
        cemployeeId.setBounds(x: 180, y: 20, width: 150, height: 20);
        add(comp: cemployeeId);
        try {
            Conn c = new Conn();
            ResultSet rs = c.s.executeQuery(string: "select * from employee");
            while(rs.next()) {
                cemployeeId.add(item: rs.getString(string: "empId"));
            }
        } catch (Exception e) {
            e.printStackTrace();
        }

        table = new JTable();

        try {
            Conn c = new Conn();
            ResultSet rs = c.s.executeQuery(string: "select * from employee");
            table.setModel(dataModel:DbUtils.resultSetToTableModel(rs));
        } catch (Exception e) {
            e.printStackTrace();
        }

        JScrollPane jsp = new JScrollPane(view: table);
        jsp.setBounds(x: 0, y: 100, width: 900, height: 600);
        add(comp: jsp);

        search = new JButton(text: "Search");
        search.setBounds(x: 20, y: 70, width: 80, height: 20);
        search.addActionListener(l: this);
```



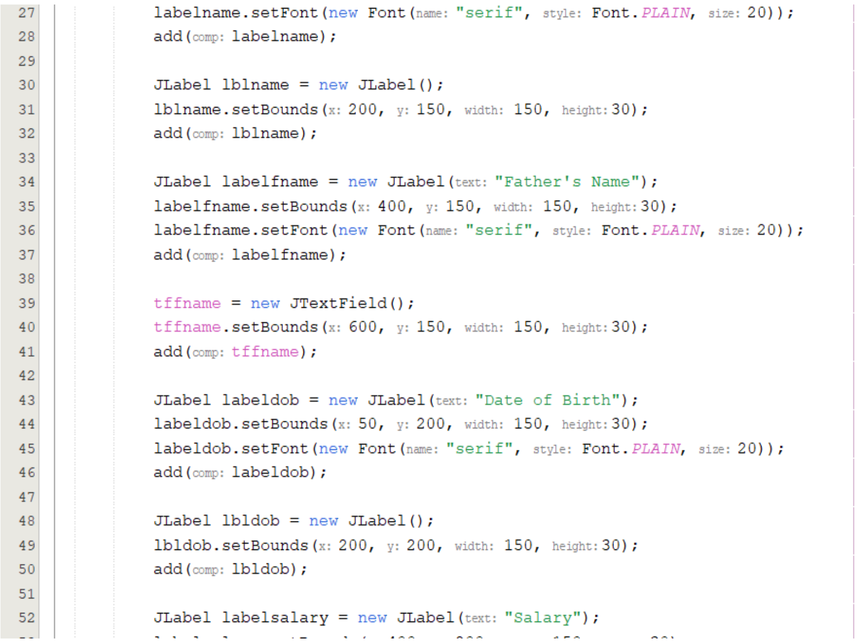
```
        add(comp: search);

        print = new JButton(text: "Print");
        print.setBounds(x: 120, y: 70, width: 80, height:20);
        print.addActionListener(l: this);
        add(comp: print);

        update = new JButton(text: "Update");
        update.setBounds(x: 220, y: 70, width: 80, height:20);
        update.addActionListener(l: this);
        add(comp: update);

        back = new JButton(text: "Back");
        back.setBounds(x: 320, y: 70, width: 80, height:20);
        back.addActionListener(l: this);
        add(comp: back);

        setSize(width: 900, height:700);
        setLocation(x: 300, y: 100);
        setVisible(b: true);
    }

    public void actionPerformed(ActionEvent ae) {
        if (ae.getSource() == search) {
            String query = "select * from employee where empId = '"+cemployeeId.getSelectedItem()+"'";
            try {
                Conn c = new Conn();
```



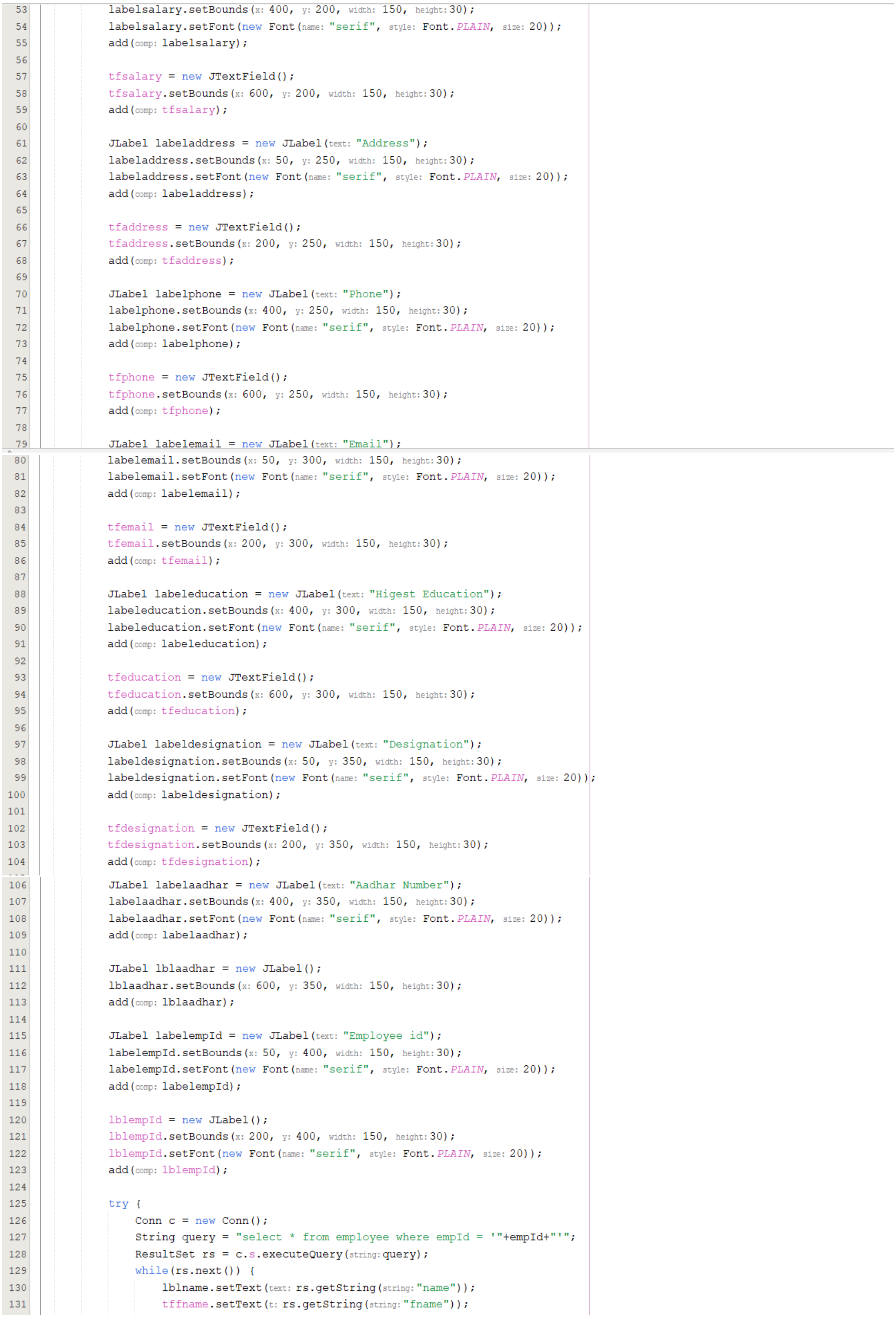
```
                ResultSet rs = c.s.executeQuery(string:query);
                table.setModel(dataModel:DbUtils.resultSetToTableModel(rs));
            } catch (Exception e) {
                e.printStackTrace();
            }
        } else if (ae.getSource() == print) {
            try {
                table.print();
            } catch (Exception e) {
                e.printStackTrace();
            }
        } else if (ae.getSource() == update) {
            setVisible(b: false);
            new UpdateEmployee(empId: cemployeeId.getSelectedItem());
        } else {
            setVisible(b: false);
            new Home();
        }
    }

    public static void main(String[] args) {
        new ViewEmployee();
    }
}
```



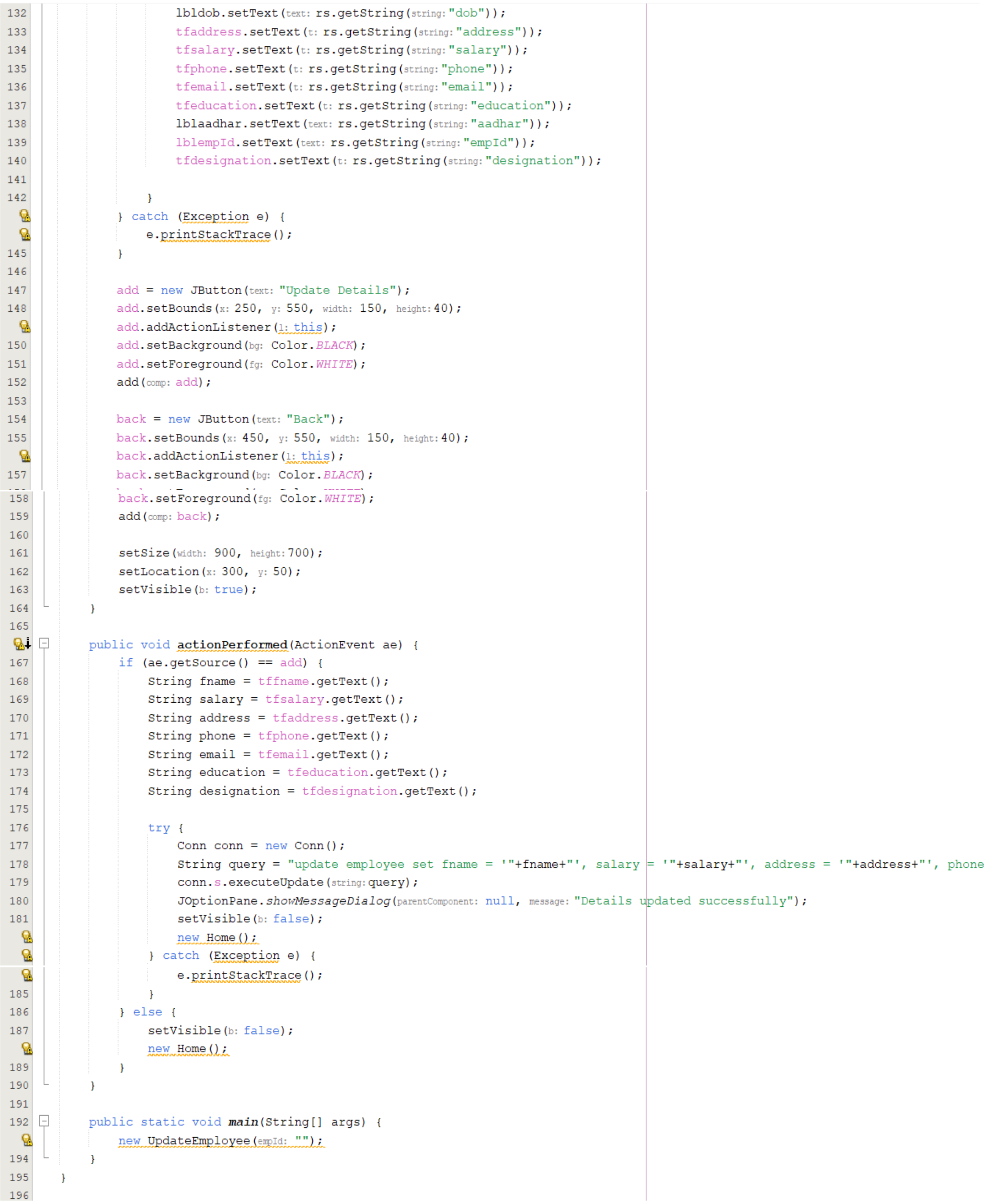
```java
package employee.management.system;

import java.awt.*;
import javax.swing.*;
import java.awt.event.*;
import java.sql.*;

public class UpdateEmployee extends JFrame implements ActionListener{

    JTextField tfeducation, tffname, tfaddress, tfphone, tfaadhar, tfemail, tfsalary, tfdesignation;
    JLabel lblempId;
    JButton add, back;
    String empId;

    UpdateEmployee(String empId) {
        this.empId = empId;
        getContentPane().setBackground(c: Color.WHITE);
        setLayout(manager: null);

        JLabel heading = new JLabel(text: "Update Employee Detail");
        heading.setBounds(x: 320, y: 30, width: 500, height:50);
        heading.setFont(new Font(name: "SAN_SERIF", style: Font.BOLD, size: 25));
        add(comp: heading);

        JLabel labelname = new JLabel(text: "Name");
        labelname.setBounds(x: 50, y: 150, width: 150, height:30);
```



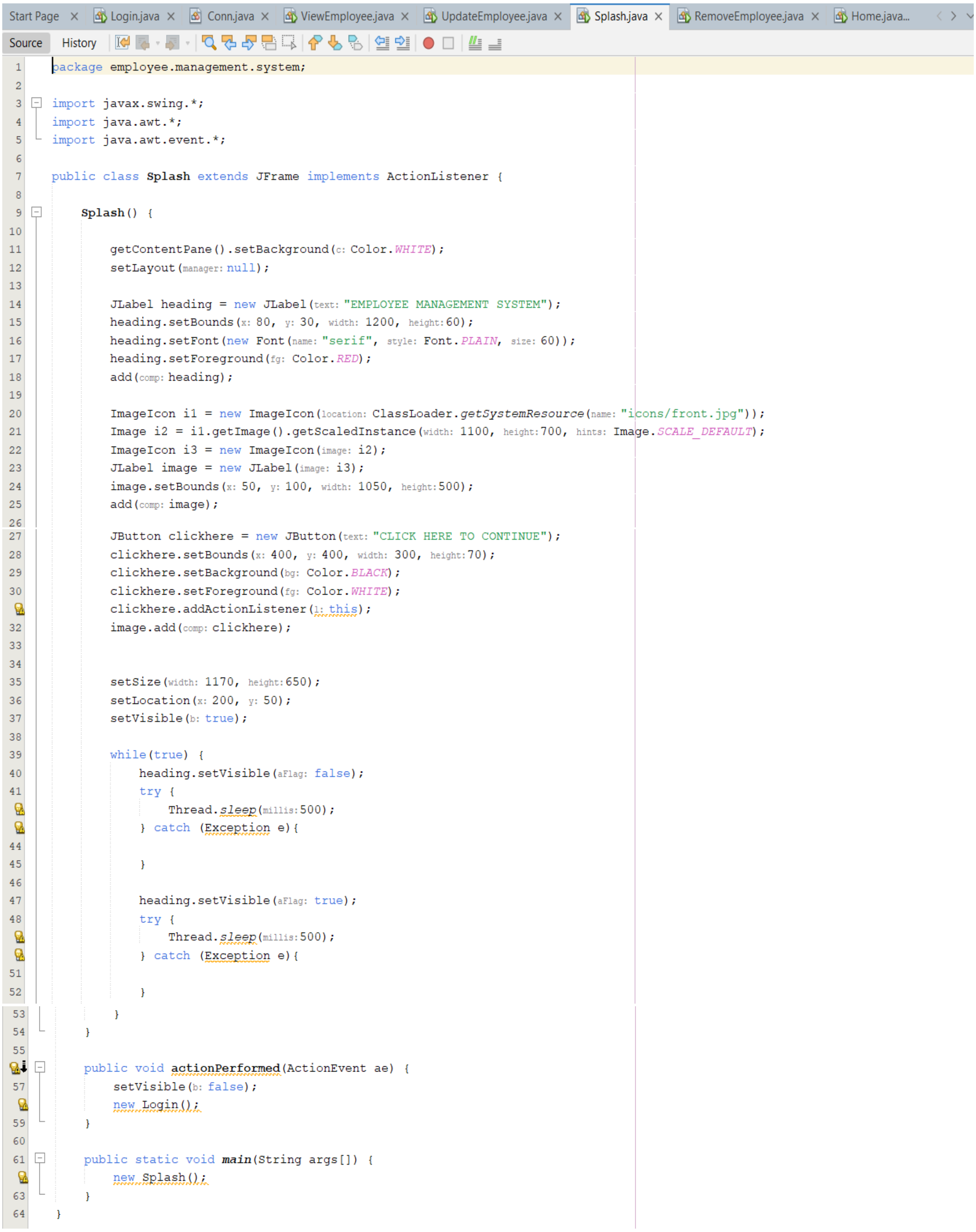
```java
        labelname.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labelname);

        JLabel lblname = new JLabel();
        lblname.setBounds(x: 200, y: 150, width: 150, height:30);
        add(comp: lblname);

        JLabel labelfname = new JLabel(text: "Father's Name");
        labelfname.setBounds(x: 400, y: 150, width: 150, height:30);
        labelfname.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labelfname);

        tffname = new JTextField();
        tffname.setBounds(x: 600, y: 150, width: 150, height:30);
        add(comp: tffname);

        JLabel labeldob = new JLabel(text: "Date of Birth");
        labeldob.setBounds(x: 50, y: 200, width: 150, height:30);
        labeldob.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labeldob);

        JLabel lbldob = new JLabel();
        lbldob.setBounds(x: 200, y: 200, width: 150, height:30);
        add(comp: lbldob);

        JLabel labelsalary = new JLabel(text: "Salary");
```



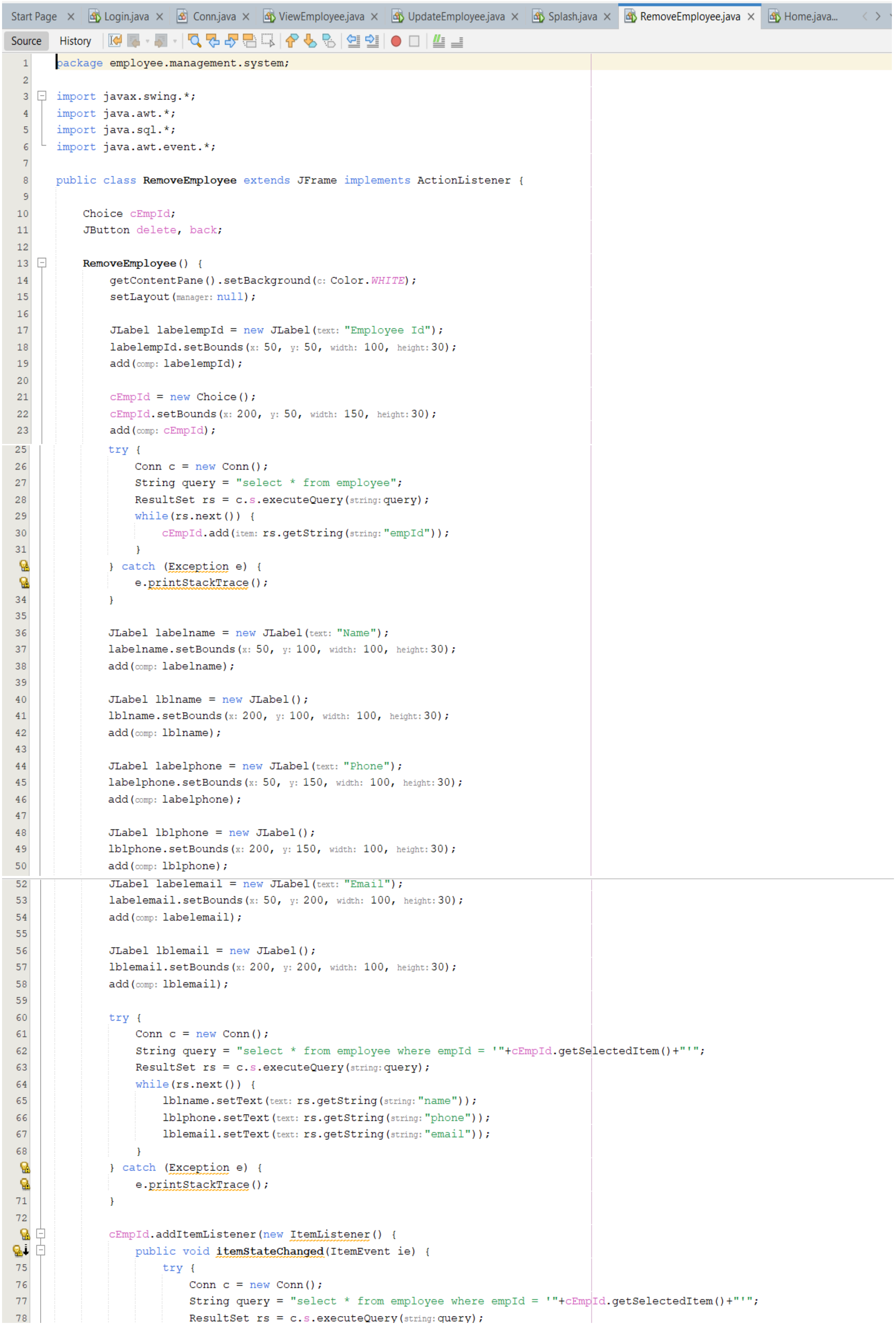
```
        labelsalary.setBounds(x: 400, y: 200, width: 150, height: 30);
        labelsalary.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labelsalary);

        tfsalary = new JTextField();
        tfsalary.setBounds(x: 600, y: 200, width: 150, height: 30);
        add(comp: tfsalary);

        JLabel labeladdress = new JLabel(text: "Address");
        labeladdress.setBounds(x: 50, y: 250, width: 150, height: 30);
        labeladdress.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labeladdress);

        tfaddress = new JTextField();
        tfaddress.setBounds(x: 200, y: 250, width: 150, height: 30);
        add(comp: tfaddress);

        JLabel labelphone = new JLabel(text: "Phone");
        labelphone.setBounds(x: 400, y: 250, width: 150, height: 30);
        labelphone.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labelphone);

        tfphone = new JTextField();
        tfphone.setBounds(x: 600, y: 250, width: 150, height: 30);
        add(comp: tfphone);

        JLabel labelemail = new JLabel(text: "Email");
        labelemail.setBounds(x: 50, y: 300, width: 150, height: 30);
        labelemail.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labelemail);

        tfemail = new JTextField();
        tfemail.setBounds(x: 200, y: 300, width: 150, height: 30);
        add(comp: tfemail);

        JLabel labeleducation = new JLabel(text: "Higest Education");
        labeleducation.setBounds(x: 400, y: 300, width: 150, height: 30);
        labeleducation.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labeleducation);

        tfeducation = new JTextField();
        tfeducation.setBounds(x: 600, y: 300, width: 150, height: 30);
        add(comp: tfeducation);

        JLabel labeldesignation = new JLabel(text: "Designation");
        labeldesignation.setBounds(x: 50, y: 350, width: 150, height: 30);
        labeldesignation.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labeldesignation);

        tfdesignation = new JTextField();
        tfdesignation.setBounds(x: 200, y: 350, width: 150, height: 30);
        add(comp: tfdesignation);
        JLabel labelaadhar = new JLabel(text: "Aadhar Number");
        labelaadhar.setBounds(x: 400, y: 350, width: 150, height: 30);
        labelaadhar.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labelaadhar);

        JLabel lblaadhar = new JLabel();
        lblaadhar.setBounds(x: 600, y: 350, width: 150, height: 30);
        add(comp: lblaadhar);

        JLabel labelempId = new JLabel(text: "Employee id");
        labelempId.setBounds(x: 50, y: 400, width: 150, height: 30);
        labelempId.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labelempId);

        lblempId = new JLabel();
        lblempId.setBounds(x: 200, y: 400, width: 150, height: 30);
        lblempId.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: lblempId);

        try {
            Conn c = new Conn();
            String query = "select * from employee where empId = '"+empId+"'";
            ResultSet rs = c.s.executeQuery(string: query);
            while(rs.next()) {
                lblname.setText(text: rs.getString(string: "name"));
                tffname.setText(t: rs.getString(string: "fname"));
```



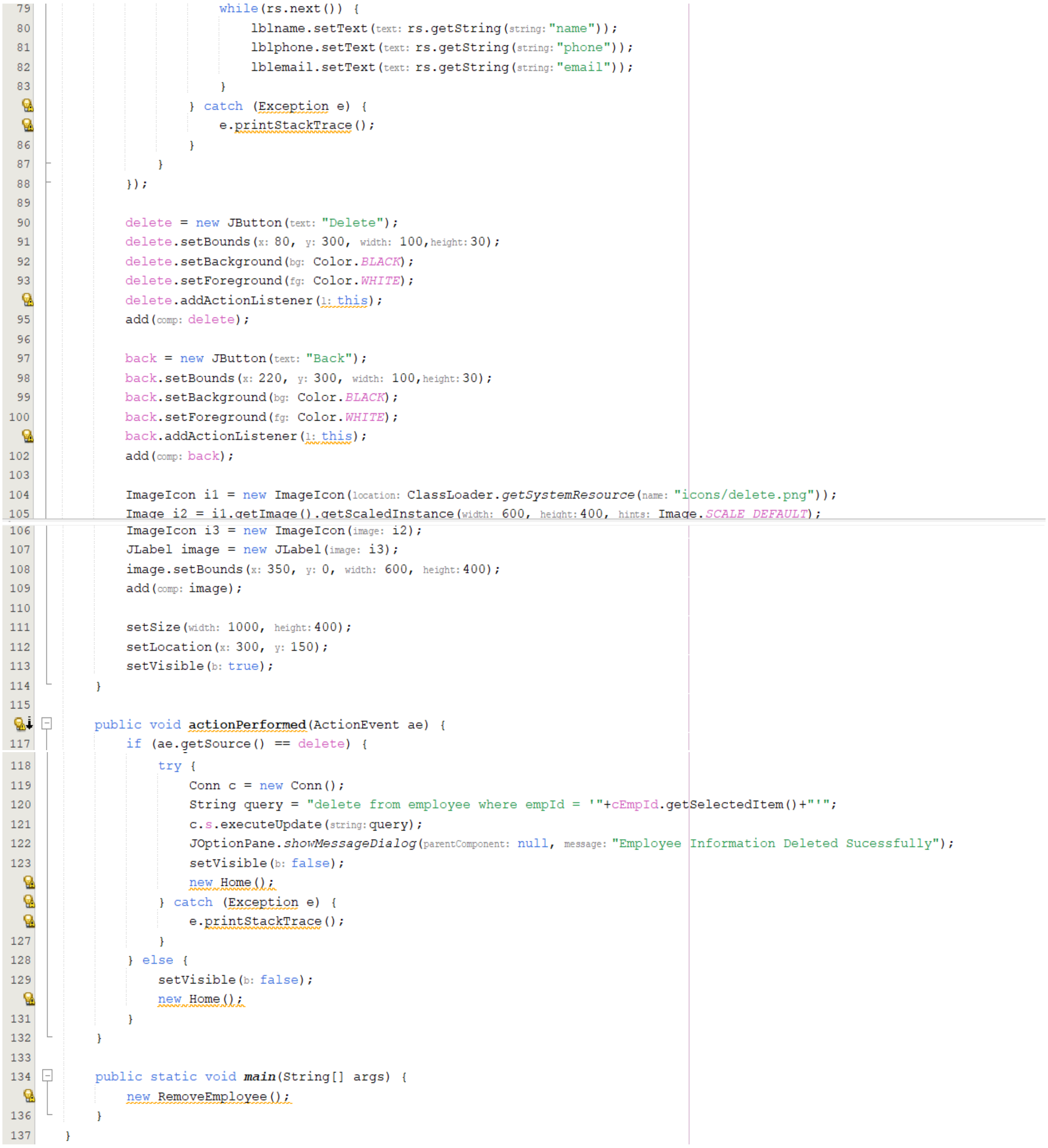
```
                lbldob.setText(text: rs.getString(string: "dob"));
                tfaddress.setText(t: rs.getString(string: "address"));
                tfsalary.setText(t: rs.getString(string: "salary"));
                tfphone.setText(t: rs.getString(string: "phone"));
                tfemail.setText(t: rs.getString(string: "email"));
                tfeducation.setText(t: rs.getString(string: "education"));
                lblaadhar.setText(text: rs.getString(string: "aadhar"));
                lblempId.setText(text: rs.getString(string: "empId"));
                tfdesignation.setText(t: rs.getString(string: "designation"));

            }
        } catch (Exception e) {
            e.printStackTrace();
        }

        add = new JButton(text: "Update Details");
        add.setBounds(x: 250, y: 550, width: 150, height: 40);
        add.addActionListener(l: this);
        add.setBackground(bg: Color.BLACK);
        add.setForeground(fg: Color.WHITE);
        add(comp: add);

        back = new JButton(text: "Back");
        back.setBounds(x: 450, y: 550, width: 150, height: 40);
        back.addActionListener(l: this);
        back.setBackground(bg: Color.BLACK);
        back.setForeground(fg: Color.WHITE);
        add(comp: back);

        setSize(width: 900, height: 700);
        setLocation(x: 300, y: 50);
        setVisible(b: true);
    }

    public void actionPerformed(ActionEvent ae) {
        if (ae.getSource() == add) {
            String fname = tffname.getText();
            String salary = tfsalary.getText();
            String address = tfaddress.getText();
            String phone = tfphone.getText();
            String email = tfemail.getText();
            String education = tfeducation.getText();
            String designation = tfdesignation.getText();

            try {
                Conn conn = new Conn();
                String query = "update employee set fname = '"+fname+"', salary = '"+salary+"', address = '"+address+"', phone
                conn.s.executeUpdate(string: query);
                JOptionPane.showMessageDialog(parentComponent: null, message: "Details updated successfully");
                setVisible(b: false);
                new Home();
            } catch (Exception e) {
                e.printStackTrace();
            }
        } else {
            setVisible(b: false);
            new Home();
        }
    }

    public static void main(String[] args) {
        new UpdateEmployee(empId: "");
    }
}
```



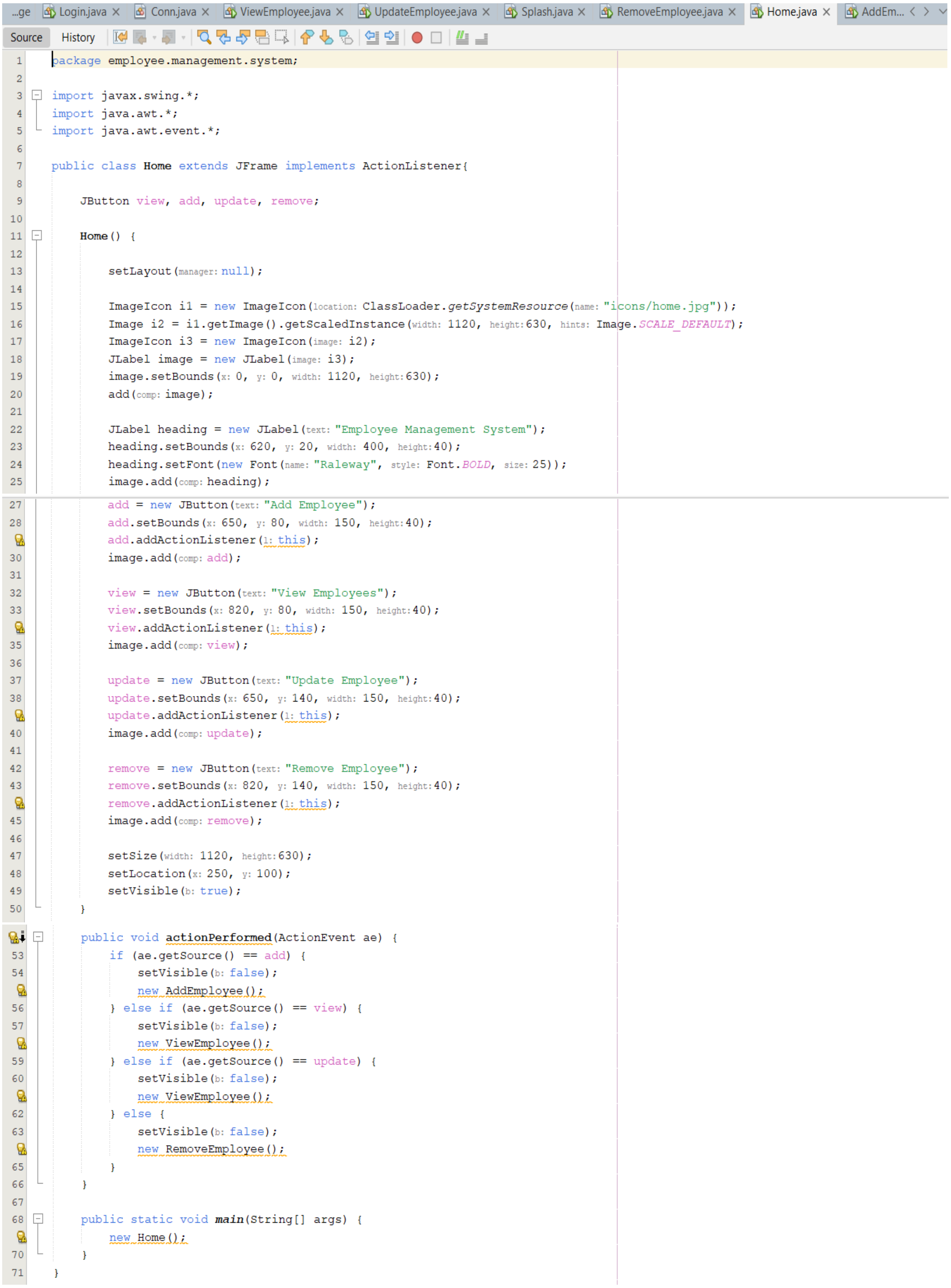
```java
package employee.management.system;

import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class Splash extends JFrame implements ActionListener {

    Splash() {

        getContentPane().setBackground(c: Color.WHITE);
        setLayout(manager: null);

        JLabel heading = new JLabel(text: "EMPLOYEE MANAGEMENT SYSTEM");
        heading.setBounds(x: 80, y: 30, width: 1200, height:60);
        heading.setFont(new Font(name: "serif", style: Font.PLAIN, size: 60));
        heading.setForeground(fg: Color.RED);
        add(comp: heading);

        ImageIcon i1 = new ImageIcon(location: ClassLoader.getSystemResource(name: "icons/front.jpg"));
        Image i2 = i1.getImage().getScaledInstance(width: 1100, height:700, hints: Image.SCALE_DEFAULT);
        ImageIcon i3 = new ImageIcon(image: i2);
        JLabel image = new JLabel(image: i3);
        image.setBounds(x: 50, y: 100, width: 1050, height:500);
        add(comp: image);

        JButton clickhere = new JButton(text: "CLICK HERE TO CONTINUE");
        clickhere.setBounds(x: 400, y: 400, width: 300, height:70);
        clickhere.setBackground(bg: Color.BLACK);
        clickhere.setForeground(fg: Color.WHITE);
        clickhere.addActionListener(l: this);
        image.add(comp: clickhere);


        setSize(width: 1170, height:650);
        setLocation(x: 200, y: 50);
        setVisible(b: true);

        while(true) {
            heading.setVisible(aFlag: false);
            try {
                Thread.sleep(millis:500);
            } catch (Exception e){

            }

            heading.setVisible(aFlag: true);
            try {
                Thread.sleep(millis:500);
            } catch (Exception e){

            }
        }
    }

    public void actionPerformed(ActionEvent ae) {
        setVisible(b: false);
        new Login();
    }

    public static void main(String args[]) {
        new Splash();
    }
}
```



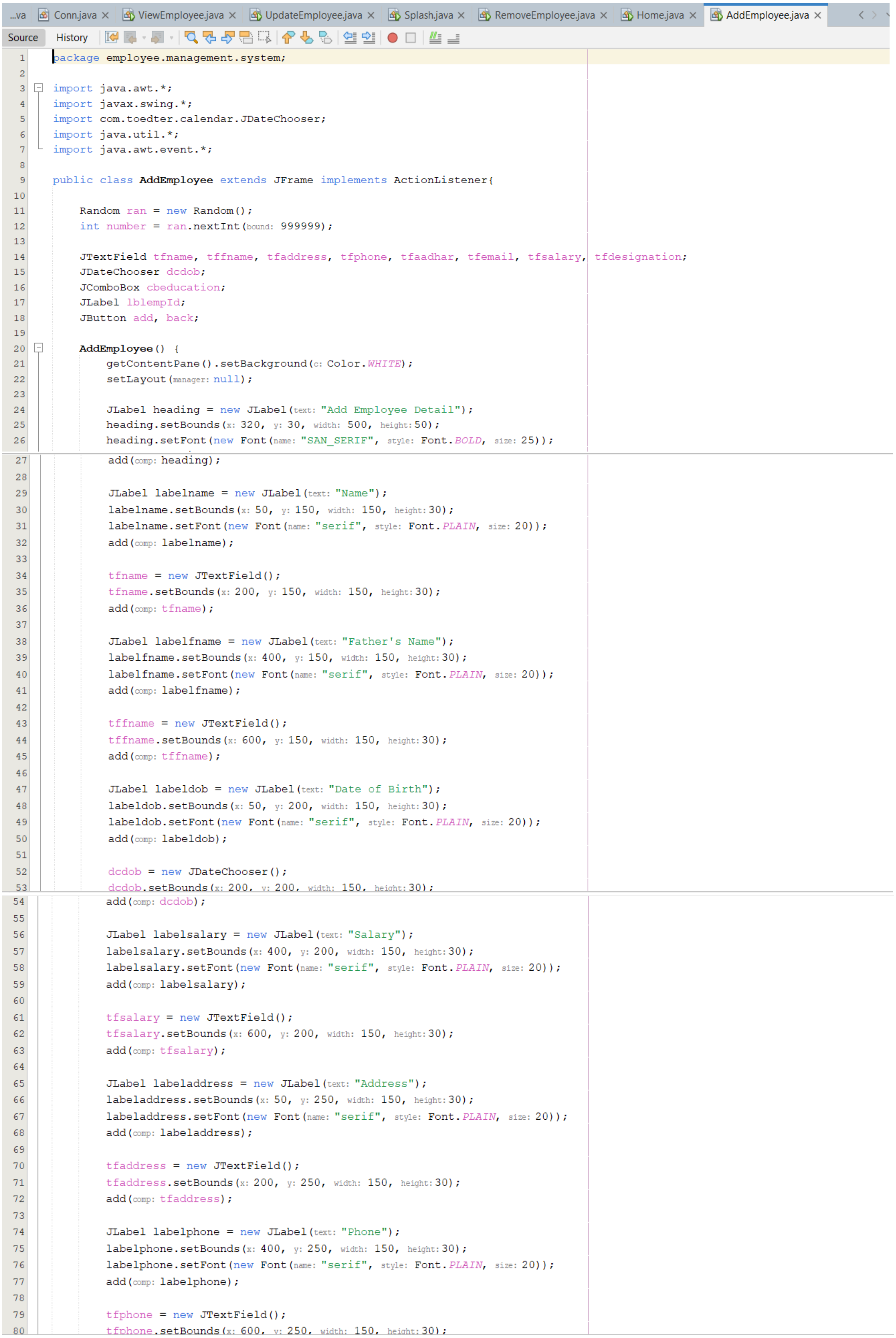
```
package employee.management.system;

import javax.swing.*;
import java.awt.*;
import java.sql.*;
import java.awt.event.*;

public class RemoveEmployee extends JFrame implements ActionListener {

    Choice cEmpId;
    JButton delete, back;

    RemoveEmployee() {
        getContentPane().setBackground(c: Color.WHITE);
        setLayout(manager: null);

        JLabel labelempId = new JLabel(text: "Employee Id");
        labelempId.setBounds(x: 50, y: 50, width: 100, height: 30);
        add(comp: labelempId);

        cEmpId = new Choice();
        cEmpId.setBounds(x: 200, y: 50, width: 150, height: 30);
        add(comp: cEmpId);
        try {
            Conn c = new Conn();
            String query = "select * from employee";
            ResultSet rs = c.s.executeQuery(string: query);
            while(rs.next()) {
                cEmpId.add(item: rs.getString(string: "empId"));
            }
        } catch (Exception e) {
            e.printStackTrace();
        }

        JLabel labelname = new JLabel(text: "Name");
        labelname.setBounds(x: 50, y: 100, width: 100, height: 30);
        add(comp: labelname);

        JLabel lblname = new JLabel();
        lblname.setBounds(x: 200, y: 100, width: 100, height: 30);
        add(comp: lblname);

        JLabel labelphone = new JLabel(text: "Phone");
        labelphone.setBounds(x: 50, y: 150, width: 100, height: 30);
        add(comp: labelphone);

        JLabel lblphone = new JLabel();
        lblphone.setBounds(x: 200, y: 150, width: 100, height: 30);
        add(comp: lblphone);
        JLabel labelemail = new JLabel(text: "Email");
        labelemail.setBounds(x: 50, y: 200, width: 100, height: 30);
        add(comp: labelemail);

        JLabel lblemail = new JLabel();
        lblemail.setBounds(x: 200, y: 200, width: 100, height: 30);
        add(comp: lblemail);

        try {
            Conn c = new Conn();
            String query = "select * from employee where empId = '"+cEmpId.getSelectedItem()+"'";
            ResultSet rs = c.s.executeQuery(string: query);
            while(rs.next()) {
                lblname.setText(text: rs.getString(string: "name"));
                lblphone.setText(text: rs.getString(string: "phone"));
                lblemail.setText(text: rs.getString(string: "email"));
            }
        } catch (Exception e) {
            e.printStackTrace();
        }

        cEmpId.addItemListener(new ItemListener() {
            public void itemStateChanged(ItemEvent ie) {
                try {
                    Conn c = new Conn();
                    String query = "select * from employee where empId = '"+cEmpId.getSelectedItem()+"'";
                    ResultSet rs = c.s.executeQuery(string: query);
```



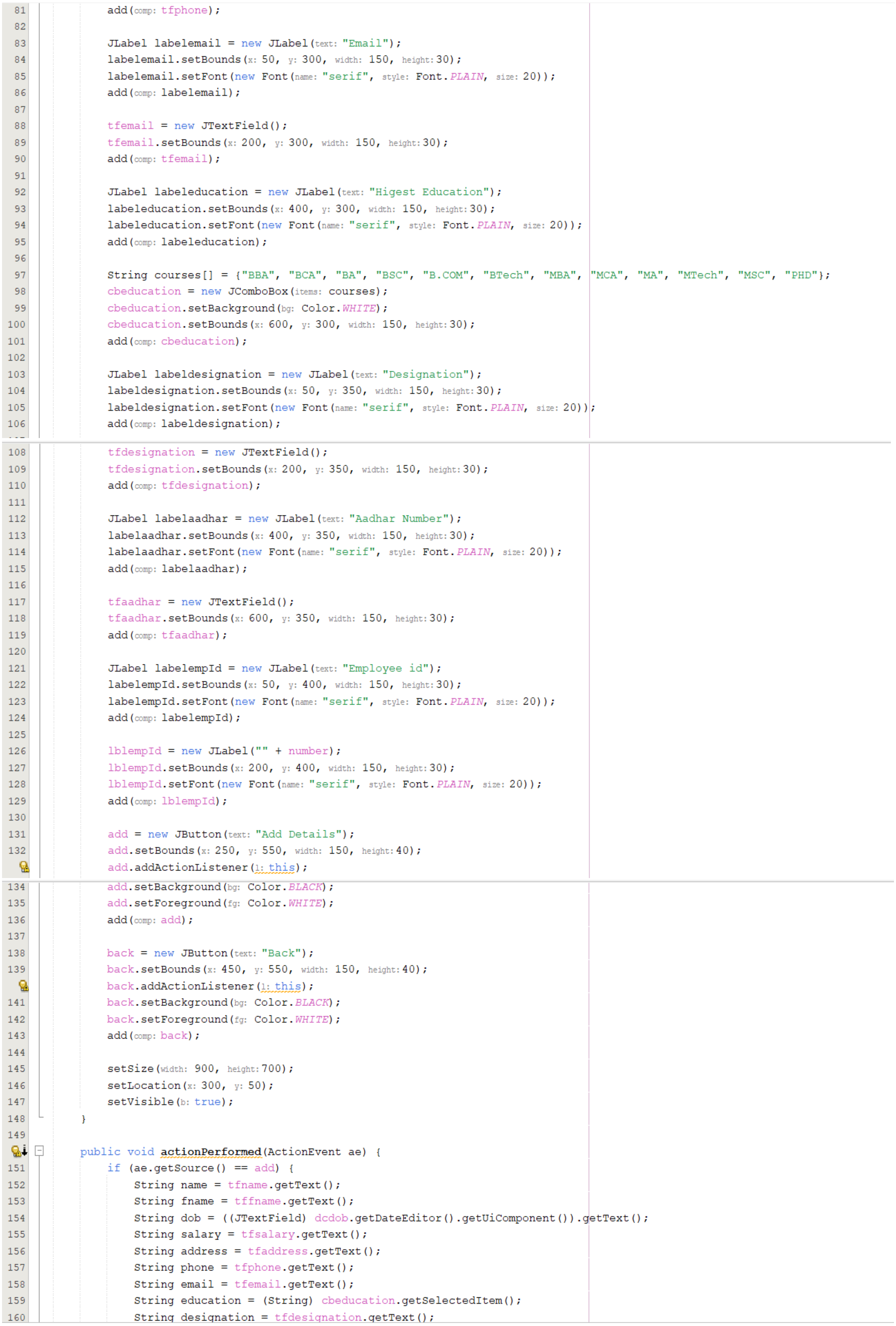
```
                    while(rs.next()) {
                        lblname.setText(text: rs.getString(string:"name"));
                        lblphone.setText(text: rs.getString(string:"phone"));
                        lblemail.setText(text: rs.getString(string:"email"));
                    }
                } catch (Exception e) {
                    e.printStackTrace();
                }
            }
        });

        delete = new JButton(text: "Delete");
        delete.setBounds(x: 80, y: 300, width: 100,height:30);
        delete.setBackground(bg: Color.BLACK);
        delete.setForeground(fg: Color.WHITE);
        delete.addActionListener(l: this);
        add(comp: delete);

        back = new JButton(text: "Back");
        back.setBounds(x: 220, y: 300, width: 100,height:30);
        back.setBackground(bg: Color.BLACK);
        back.setForeground(fg: Color.WHITE);
        back.addActionListener(l: this);
        add(comp: back);

        ImageIcon i1 = new ImageIcon(location: ClassLoader.getSystemResource(name: "icons/delete.png"));
        Image i2 = i1.getImage().getScaledInstance(width: 600, height:400, hints: Image.SCALE_DEFAULT);
        ImageIcon i3 = new ImageIcon(image: i2);
        JLabel image = new JLabel(image: i3);
        image.setBounds(x: 350, y: 0, width: 600, height:400);
        add(comp: image);

        setSize(width: 1000, height:400);
        setLocation(x: 300, y: 150);
        setVisible(b: true);
    }

    public void actionPerformed(ActionEvent ae) {
        if (ae.getSource() == delete) {
            try {
                Conn c = new Conn();
                String query = "delete from employee where empId = '"+cEmpId.getSelectedItem()+"'";
                c.s.executeUpdate(string:query);
                JOptionPane.showMessageDialog(parentComponent: null, message: "Employee Information Deleted Sucessfully");
                setVisible(b: false);
                new Home();
            } catch (Exception e) {
                e.printStackTrace();
            }
        } else {
            setVisible(b: false);
            new Home();
        }
    }

    public static void main(String[] args) {
        new RemoveEmployee();
    }
}
```



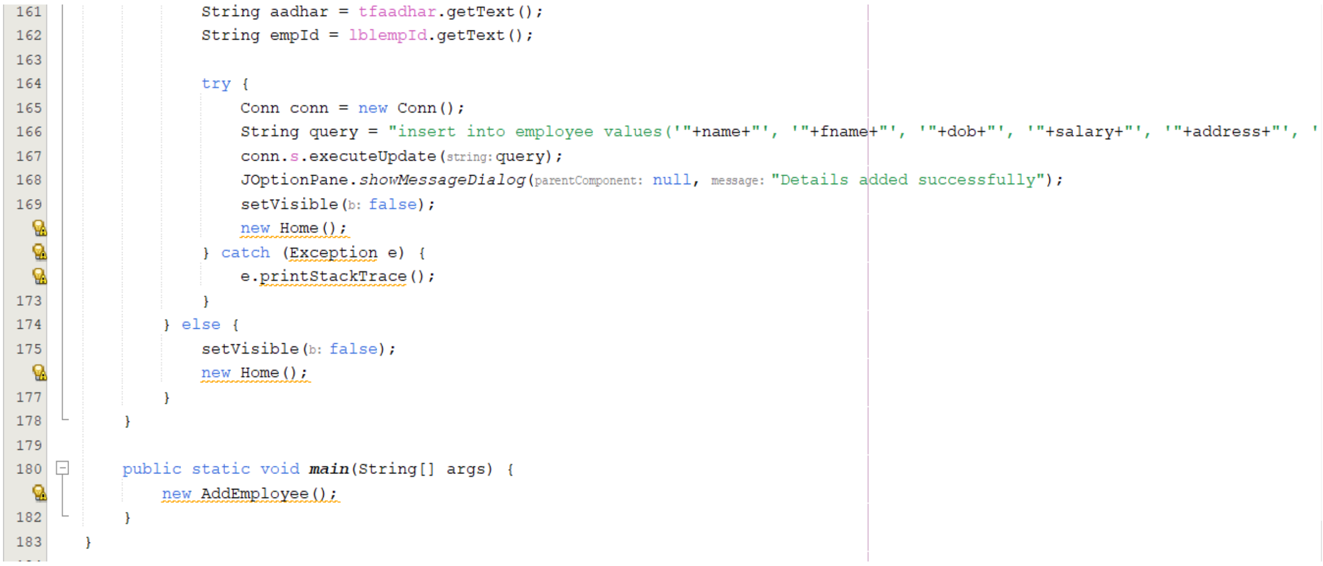
```
package employee.management.system;

import javax.swing.*;
import java.awt.*;
import java.awt.event.*;

public class Home extends JFrame implements ActionListener{

    JButton view, add, update, remove;

    Home() {

        setLayout(manager: null);

        ImageIcon i1 = new ImageIcon(location: ClassLoader.getSystemResource(name: "icons/home.jpg"));
        Image i2 = i1.getImage().getScaledInstance(width: 1120, height:630, hints: Image.SCALE_DEFAULT);
        ImageIcon i3 = new ImageIcon(image: i2);
        JLabel image = new JLabel(image: i3);
        image.setBounds(x: 0, y: 0, width: 1120, height:630);
        add(comp: image);

        JLabel heading = new JLabel(text: "Employee Management System");
        heading.setBounds(x: 620, y: 20, width: 400, height:40);
        heading.setFont(new Font(name: "Raleway", style: Font.BOLD, size: 25));
        image.add(comp: heading);
        add = new JButton(text: "Add Employee");
        add.setBounds(x: 650, y: 80, width: 150, height:40);
        add.addActionListener(l: this);
        image.add(comp: add);

        view = new JButton(text: "View Employees");
        view.setBounds(x: 820, y: 80, width: 150, height:40);
        view.addActionListener(l: this);
        image.add(comp: view);

        update = new JButton(text: "Update Employee");
        update.setBounds(x: 650, y: 140, width: 150, height:40);
        update.addActionListener(l: this);
        image.add(comp: update);

        remove = new JButton(text: "Remove Employee");
        remove.setBounds(x: 820, y: 140, width: 150, height:40);
        remove.addActionListener(l: this);
        image.add(comp: remove);

        setSize(width: 1120, height:630);
        setLocation(x: 250, y: 100);
        setVisible(b: true);
    }

    public void actionPerformed(ActionEvent ae) {
        if (ae.getSource() == add) {
            setVisible(b: false);
            new AddEmployee();
        } else if (ae.getSource() == view) {
            setVisible(b: false);
            new ViewEmployee();
        } else if (ae.getSource() == update) {
            setVisible(b: false);
            new ViewEmployee();
        } else {
            setVisible(b: false);
            new RemoveEmployee();
        }
    }

    public static void main(String[] args) {
        new Home();
    }
}
```



```java
package employee.management.system;

import java.awt.*;
import javax.swing.*;
import com.toedter.calendar.JDateChooser;
import java.util.*;
import java.awt.event.*;

public class AddEmployee extends JFrame implements ActionListener{

    Random ran = new Random();
    int number = ran.nextInt(bound: 999999);

    JTextField tfname, tffname, tfaddress, tfphone, tfaadhar, tfemail, tfsalary, tfdesignation;
    JDateChooser dcdob;
    JComboBox cbeducation;
    JLabel lblempId;
    JButton add, back;

    AddEmployee() {
        getContentPane().setBackground(c: Color.WHITE);
        setLayout(manager: null);

        JLabel heading = new JLabel(text: "Add Employee Detail");
        heading.setBounds(x: 320, y: 30, width: 500, height: 50);
        heading.setFont(new Font(name: "SAN_SERIF", style: Font.BOLD, size: 25));
        add(comp: heading);

        JLabel labelname = new JLabel(text: "Name");
        labelname.setBounds(x: 50, y: 150, width: 150, height: 30);
        labelname.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labelname);

        tfname = new JTextField();
        tfname.setBounds(x: 200, y: 150, width: 150, height: 30);
        add(comp: tfname);

        JLabel labelfname = new JLabel(text: "Father's Name");
        labelfname.setBounds(x: 400, y: 150, width: 150, height: 30);
        labelfname.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labelfname);

        tffname = new JTextField();
        tffname.setBounds(x: 600, y: 150, width: 150, height: 30);
        add(comp: tffname);

        JLabel labeldob = new JLabel(text: "Date of Birth");
        labeldob.setBounds(x: 50, y: 200, width: 150, height: 30);
        labeldob.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labeldob);

        dcdob = new JDateChooser();
        dcdob.setBounds(x: 200, y: 200, width: 150, height: 30);
        add(comp: dcdob);

        JLabel labelsalary = new JLabel(text: "Salary");
        labelsalary.setBounds(x: 400, y: 200, width: 150, height: 30);
        labelsalary.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labelsalary);

        tfsalary = new JTextField();
        tfsalary.setBounds(x: 600, y: 200, width: 150, height: 30);
        add(comp: tfsalary);

        JLabel labeladdress = new JLabel(text: "Address");
        labeladdress.setBounds(x: 50, y: 250, width: 150, height: 30);
        labeladdress.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labeladdress);

        tfaddress = new JTextField();
        tfaddress.setBounds(x: 200, y: 250, width: 150, height: 30);
        add(comp: tfaddress);

        JLabel labelphone = new JLabel(text: "Phone");
        labelphone.setBounds(x: 400, y: 250, width: 150, height: 30);
        labelphone.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labelphone);

        tfphone = new JTextField();
        tfphone.setBounds(x: 600, y: 250, width: 150, height: 30);
```



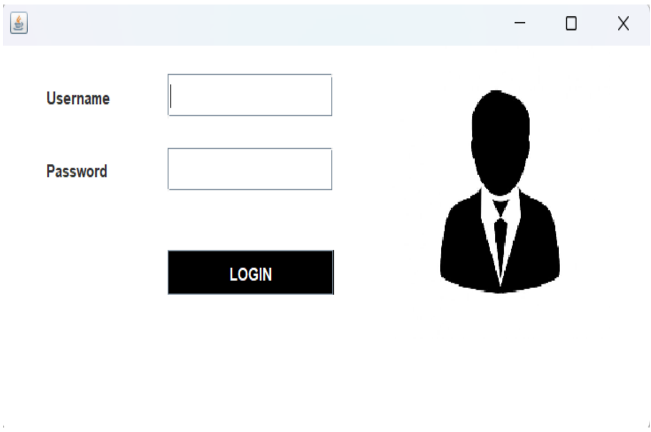
```
        add(comp: tfphone);

        JLabel labelemail = new JLabel(text: "Email");
        labelemail.setBounds(x: 50, y: 300, width: 150, height: 30);
        labelemail.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labelemail);

        tfemail = new JTextField();
        tfemail.setBounds(x: 200, y: 300, width: 150, height: 30);
        add(comp: tfemail);

        JLabel labeleducation = new JLabel(text: "Higest Education");
        labeleducation.setBounds(x: 400, y: 300, width: 150, height: 30);
        labeleducation.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labeleducation);

        String courses[] = {"BBA", "BCA", "BA", "BSC", "B.COM", "BTech", "MBA", "MCA", "MA", "MTech", "MSC", "PHD"};
        cbeducation = new JComboBox(items: courses);
        cbeducation.setBackground(bg: Color.WHITE);
        cbeducation.setBounds(x: 600, y: 300, width: 150, height: 30);
        add(comp: cbeducation);

        JLabel labeldesignation = new JLabel(text: "Designation");
        labeldesignation.setBounds(x: 50, y: 350, width: 150, height: 30);
        labeldesignation.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labeldesignation);

        tfdesignation = new JTextField();
        tfdesignation.setBounds(x: 200, y: 350, width: 150, height: 30);
        add(comp: tfdesignation);

        JLabel labelaadhar = new JLabel(text: "Aadhar Number");
        labelaadhar.setBounds(x: 400, y: 350, width: 150, height: 30);
        labelaadhar.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labelaadhar);

        tfaadhar = new JTextField();
        tfaadhar.setBounds(x: 600, y: 350, width: 150, height: 30);
        add(comp: tfaadhar);

        JLabel labelempId = new JLabel(text: "Employee id");
        labelempId.setBounds(x: 50, y: 400, width: 150, height: 30);
        labelempId.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: labelempId);

        lblempId = new JLabel("" + number);
        lblempId.setBounds(x: 200, y: 400, width: 150, height: 30);
        lblempId.setFont(new Font(name: "serif", style: Font.PLAIN, size: 20));
        add(comp: lblempId);

        add = new JButton(text: "Add Details");
        add.setBounds(x: 250, y: 550, width: 150, height: 40);
        add.addActionListener(l: this);
        add.setBackground(bg: Color.BLACK);
        add.setForeground(fg: Color.WHITE);
        add(comp: add);

        back = new JButton(text: "Back");
        back.setBounds(x: 450, y: 550, width: 150, height: 40);
        back.addActionListener(l: this);
        back.setBackground(bg: Color.BLACK);
        back.setForeground(fg: Color.WHITE);
        add(comp: back);

        setSize(width: 900, height: 700);
        setLocation(x: 300, y: 50);
        setVisible(b: true);
    }

    public void actionPerformed(ActionEvent ae) {
        if (ae.getSource() == add) {
            String name = tfname.getText();
            String fname = tffname.getText();
            String dob = ((JTextField) dcdob.getDateEditor().getUiComponent()).getText();
            String salary = tfsalary.getText();
            String address = tfaddress.getText();
            String phone = tfphone.getText();
            String email = tfemail.getText();
            String education = (String) cbeducation.getSelectedItem();
            String designation = tfdesignation.getText();
```



```
            String aadhar = tfaadhar.getText();
            String empId = lblempId.getText();

            try {
                Conn conn = new Conn();
                String query = "insert into employee values('"+name+"', '"+fname+"', '"+dob+"', '"+salary+"', '"+address+"', '
                conn.s.executeUpdate(string: query);
                JOptionPane.showMessageDialog(parentComponent: null, message: "Details added successfully");
                setVisible(b: false);
                new Home();
            } catch (Exception e) {
                e.printStackTrace();
            }
        } else {
            setVisible(b: false);
            new Home();
        }
    }

    public static void main(String[] args) {
        new AddEmployee();
    }
}
```

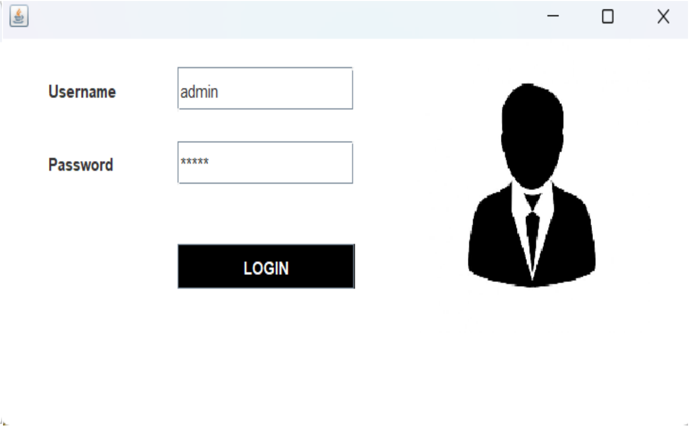
Output

Home Page:



Username

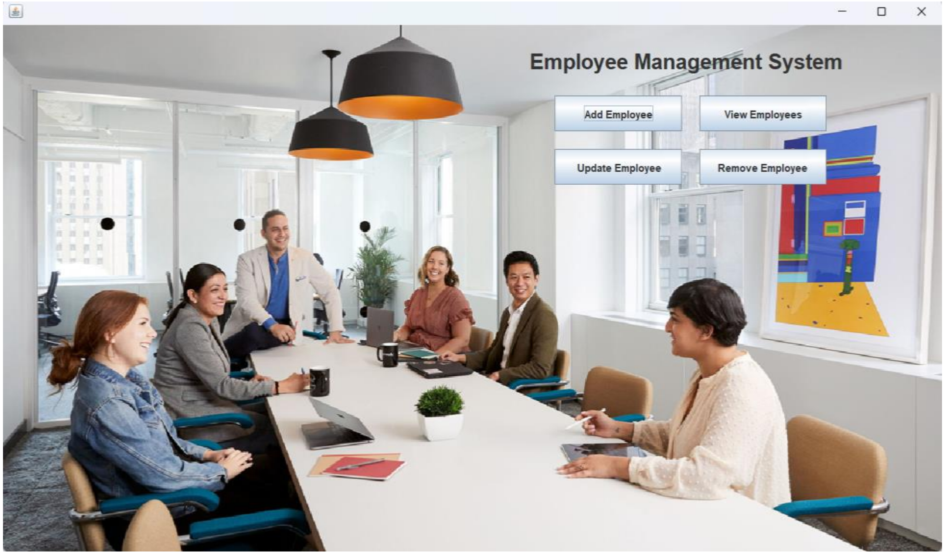
Password



Username admin

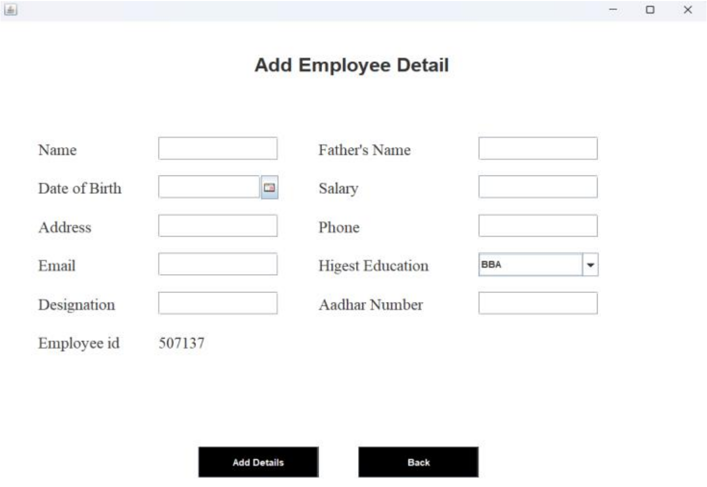
Password *****

Operations:



Employee Management System

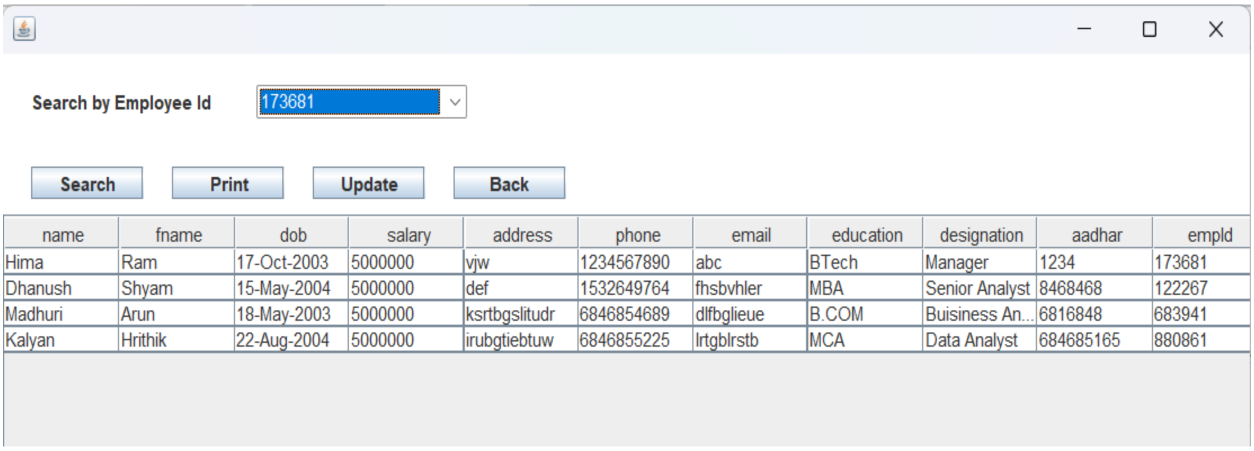
Add Employee:



Add Employee Detail

Name		Father's Name	
Date of Birth		Salary	
Address		Phone	
Email		Higest Education	BBA
Designation		Aadhar Number	
Employee id	507137		

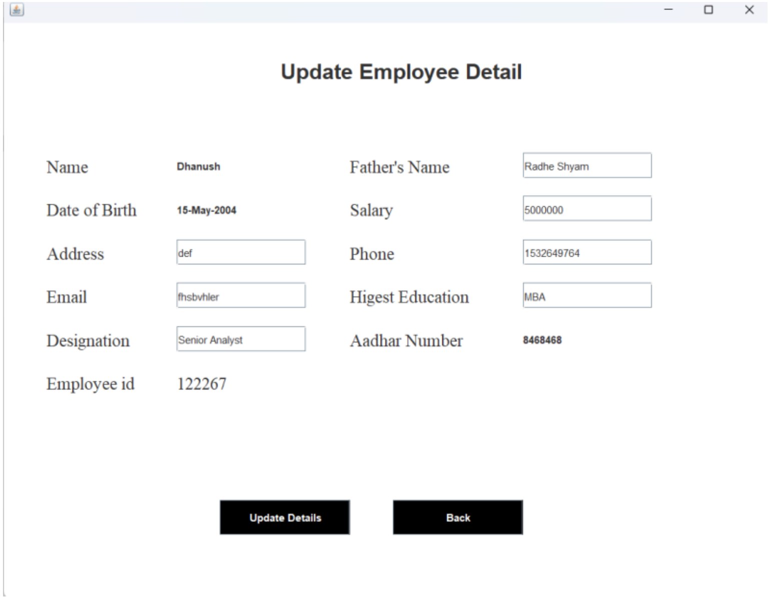
View Employee:



Search by Employee Id: 173681

name	fname	dob	salary	address	phone	email	education	designation	aadhar	empld
Hima	Ram	17-Oct-2003	5000000	vjw	1234567890	abc	BTech	Manager	1234	173681
Dhanush	Shyam	15-May-2004	5000000	def	1532649764	fhsbvhler	MBA	Senior Analyst	8468468	122267
Madhuri	Arun	18-May-2003	5000000	ksrtbgslitudr	6846854689	dlfbglieue	B.COM	Buisiness An...	6816848	683941
Kalyan	Hrithik	22-Aug-2004	5000000	irubgtiebtuw	6846855225	lrtgblrstb	MCA	Data Analyst	684685165	880861

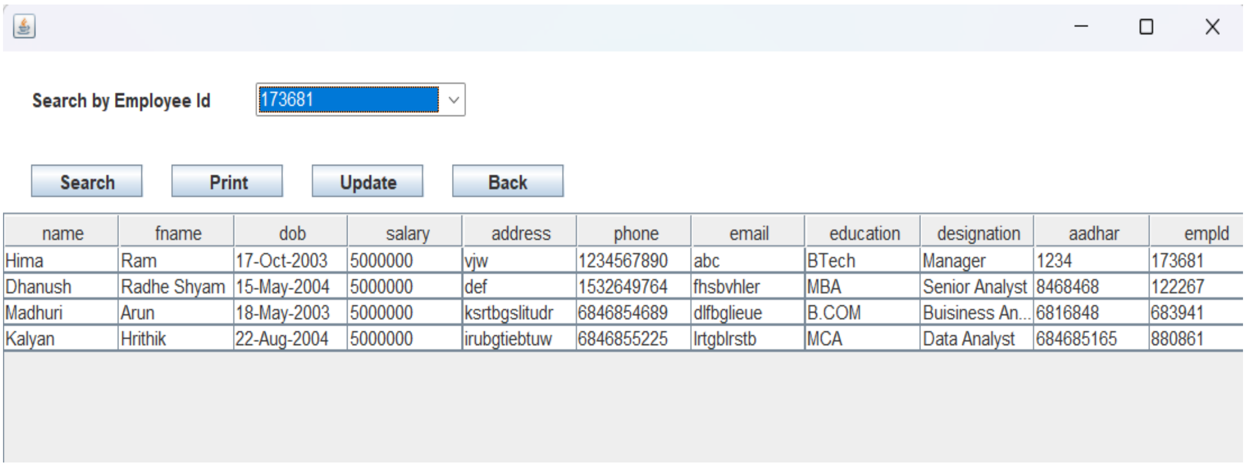
Update Employee:



Update Employee Detail

Name	Dhanush	Father's Name	Radhe Shyam
Date of Birth	15-May-2004	Salary	5000000
Address	def	Phone	1532649764
Email	fhsbvhler	Higest Education	MBA
Designation	Senior Analyst	Aadhar Number	8468468
Employee id	122267		

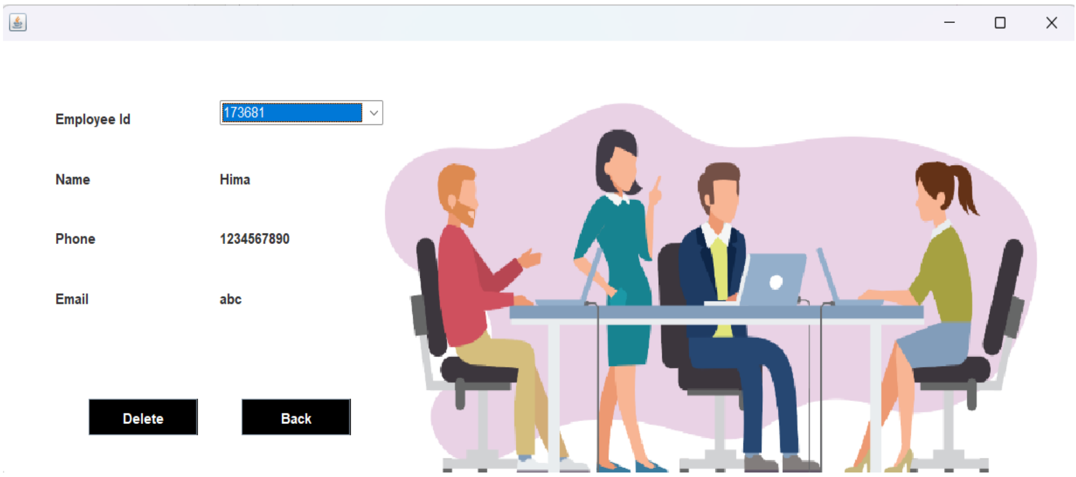
Successful Updation:



Search by Employee Id: 173681

name	fname	dob	salary	address	phone	email	education	designation	aadhar	empld
Hima	Ram	17-Oct-2003	5000000	vjw	1234567890	abc	BTech	Manager	1234	173681
Dhanush	Radhe Shyam	15-May-2004	5000000	def	1532649764	fhsbvhler	MBA	Senior Analyst	8468468	122267
Madhuri	Arun	18-May-2003	5000000	ksrtbgslitudr	6846854689	dlfbglieue	B.COM	Buisiness An...	6816848	683941
Kalyan	Hrithik	22-Aug-2004	5000000	irubgtiebtuw	6846855225	lrtgblrstb	MCA	Data Analyst	684685165	880861

Delete Employee:

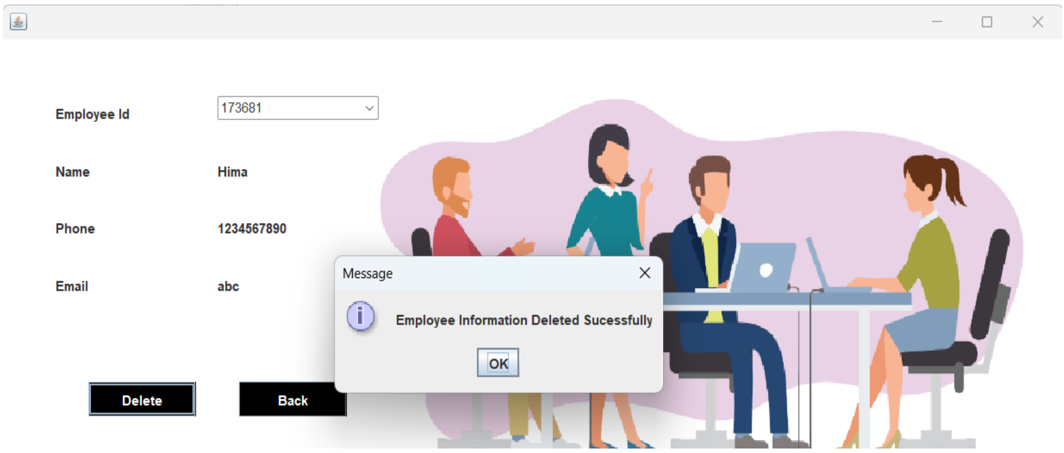


Employee Id: 173681

Name: Hima

Phone: 1234567890

Email: abc



Employee Id: 173681

Name: Hima

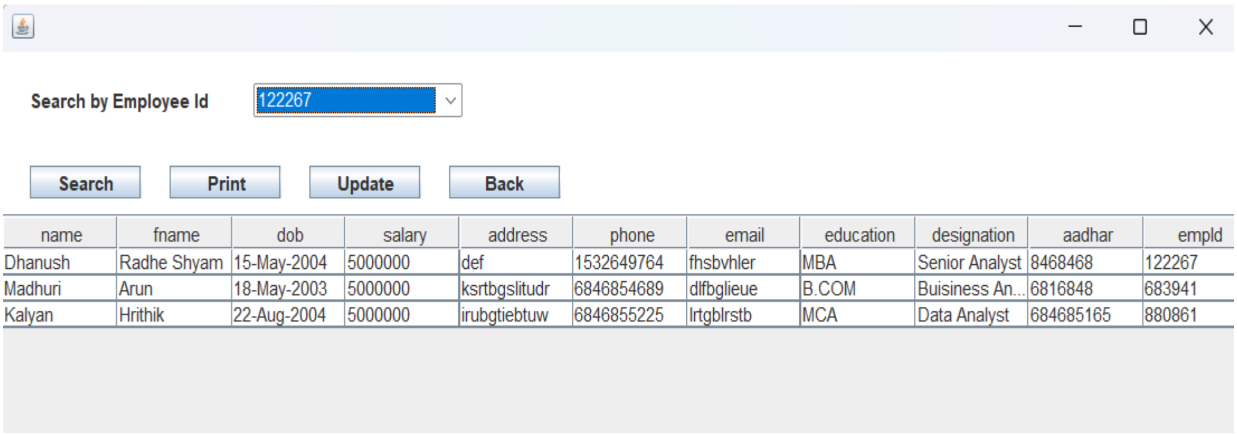
Phone: 1234567890

Email: abc

Message

Employee Information Deleted Sucessfully

Successful Deletion:



Search by Employee Id: 122267

name	fname	dob	salary	address	phone	email	education	designation	aadhar	empId
Dhanush	Radhe Shyam	15-May-2004	5000000	def	1532649764	fhsbvhler	MBA	Senior Analyst	8468468	122267
Madhuri	Arun	18-May-2003	5000000	ksrtbgslitudr	6846854689	dlfbglieue	B.COM	Buisiness An...	6816848	683941
Kalyan	Hrithik	22-Aug-2004	5000000	irubgtiebtuw	6846855225	lrtgblrstb	MCA	Data Analyst	684685165	880861

Explanation of the Source code:

Login class:

- Takes the input of the username and password

Connection class:

- Connects the database and Java compiler

View Employee class:

- Displays the details of all the employees present

Update Employee class:

- Takes the input of Employee Id, details that require changes, and updates them

Remove Employee class:

- Takes the input of Employee Id and deletes the respective employee details

Home class:

- It creates the home page which displays the operations that can be performed (Add Employee, View Employee, Update Employee, Delete Employee)

Add Employee class:

- It takes the details of an employee and adds the employee to the existing employee list

Conclusion

- This project has been a great experience for all of us as it developed our knowledge of JAVA programming and helped us improve our programming and creativity skills.

- We used the concept of classes, inheritance, and interfaces to develop this project which is interesting as well as useful.

- We would like to thank our guide, Sudhakar sir, for giving us the proper guidance which was much needed.
We hope, our project met your expectations and that our efforts are reflected in this project.