**MySQL steps**

**Create database**

create database ems;

**Select the database**

use ems;

**Create employee table**

create table employee (  
 id int primary key,  
 name varchar(25),  
 gender varchar(10),  
 phoneNum varchar(13),  
 email varchar(25),  
 designation varchar(20),  
 salary double  
);

**Create admin table**

create table admin (  
 username varchar(25) primary key,  
 password varchar(25)  
);

**Insert some values in the admin table for login**

insert into admin values  
("admin","admin123"),  
("admin2","0000");

**Insert values into employees (optional)**

insert into employee values

(142,"Jon Snow","Male","8454562158","jon@gmail.com","Associate",30000.0),

(127,"Robb Stark","Male","7654812345","robb@gmail.com","Manager",80000.0);

**Employee Management System Project in Java**

**1. Importing required libraries**

import java.awt.\*;

import java.awt.event.KeyAdapter;

import java.awt.event.KeyEvent;

import java.sql.\*;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

import javax.swing.\*;

import javax.swing.border.EmptyBorder;

**2. Create a connection with SQL in Java**

Give the name of your database, username, and password in the below fields.

// gets the required driver

Class.forName("com.mysql.cj.jdbc.Driver");

// creates connection with sql

String dbName = "ems";

String db\_username = "root";

String db\_password = "master";

Connection con= DriverManager.getConnection(

"jdbc:mysql://localhost:3306/"+dbName, db\_username, db\_password);

Create an instance of the employee management system and pass to it the connection object so that it can later be used.

EmployeeManagementSystem ems = new EmployeeManagementSystem(con);

**3. Admin Login for employee management system in java**

Users should be able to log in with their credentials. We are also going to show popups on invalid usernames or passwords.

First, we will create the login screen by using JFrame and add all components to it.

ems.loginFrame = new JFrame();

JLabel usernameLabel = new JLabel("Username");

usernameLabel.setBounds(200,150,220,50);

usernameLabel.setFont(new Font("Times New Roman", Font.*PLAIN*, 30));

ems.loginFrame.add(usernameLabel);

JTextField usernameField = new JTextField();

usernameField.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));

usernameField.setBounds(450,150,420,50);

ems.loginFrame.add(usernameField);

JLabel passwordLabel = new JLabel("Password");

passwordLabel.setFont(new Font("Times New Roman", Font.*PLAIN*, 30));

passwordLabel.setBounds(200,250,220,50);

ems.loginFrame.add(passwordLabel);

JPasswordField passwordField = new JPasswordField();

passwordField.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));

passwordField.setBounds(450,250,420,50);

ems.loginFrame.add(passwordField);

JButton submitButton=new JButton("Submit");

submitButton.setFont(new Font("Times New Roman", Font.*PLAIN*, 30));

submitButton.setBounds(450,400,250, 50);

ems.loginFrame.add(submitButton);

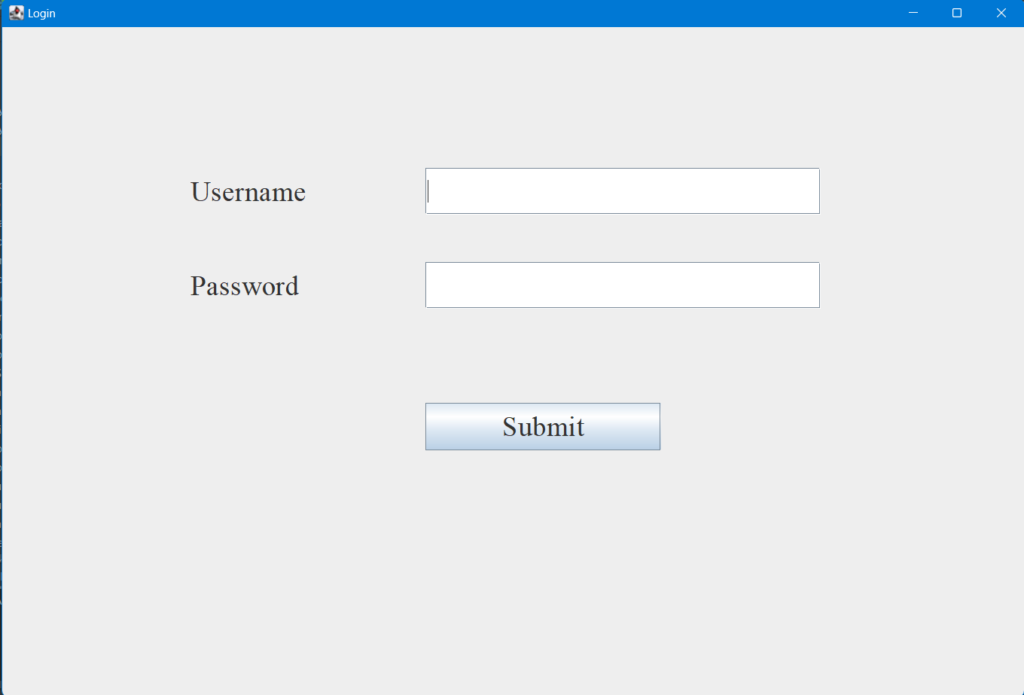
ems.loginFrame.setSize(1100,750);

ems.loginFrame.setLayout(null);

ems.loginFrame.setVisible(true);

ems.loginFrame.setDefaultCloseOperation(WindowConstants.EXIT\_ON\_CLOSE);

We get the below screen with this code



Now let’s add functionalities to it

We will create an action listener on the submit button. This will allow us to get the user input and validate it for authentication

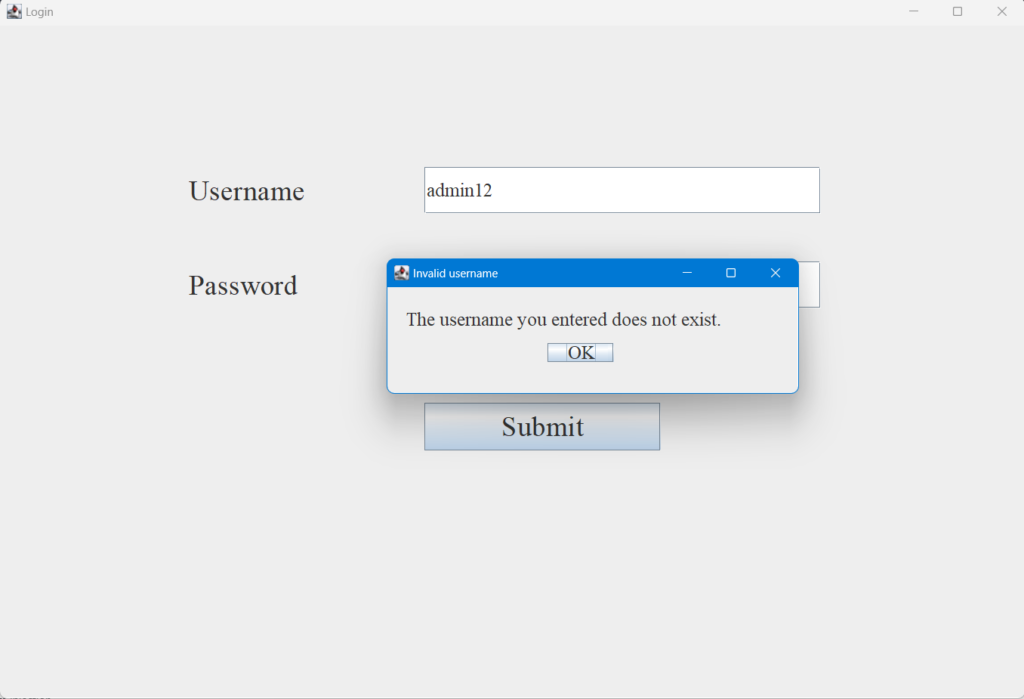
submitButton.addActionListener(actionEvent -> {  
 *username* = usernameField.getText();  
 *password* = String.*valueOf*(passwordField.getPassword());  
  
 try {  
 boolean auth = ems.login(*username*, *password*);  
 if(auth){  
 ems.loginFrame.dispose();  
 ems.mainMenu();  
 }  
  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
});

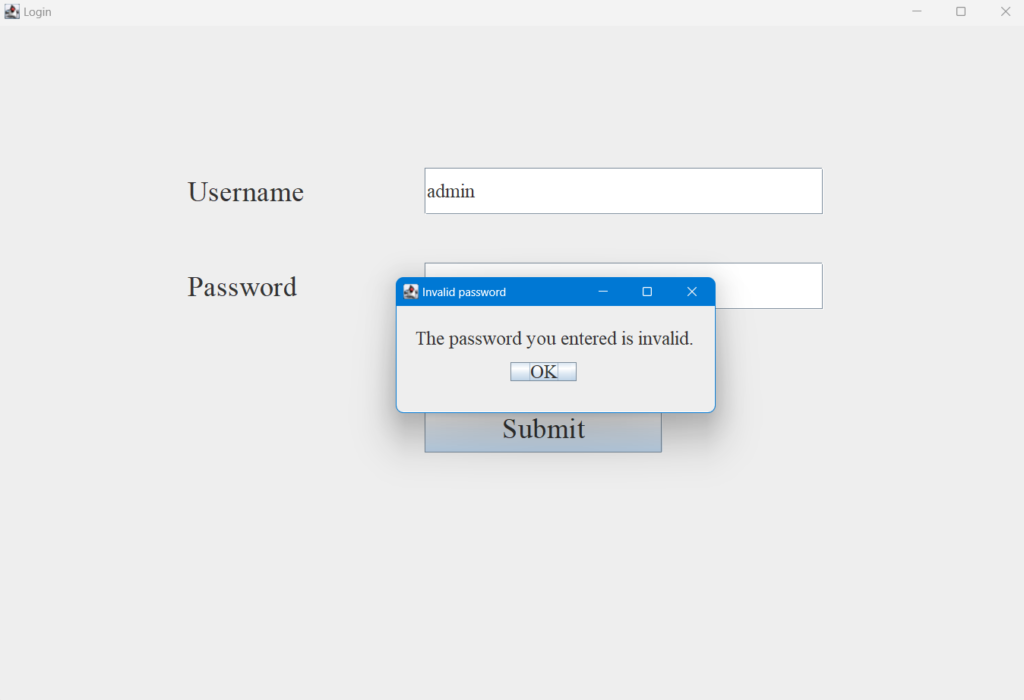
**4. Login screen for employee management system java project**

Now let’s write the login functionality

public boolean login(String username, String password) throws SQLException {  
 Statement statement = this.con.createStatement();  
 String q = String.*format*("select password from admin where username = '%s';", username);  
 ResultSet resultSet = statement.executeQuery(q);  
// gets us password if the username exists  
 if(resultSet.next()){  
// compare password with user input  
 if(resultSet.getString(1).equals(password)){  
 return true;  
 }  
 else {  
 JFrame popup = new JFrame("Invalid password");  
 JLabel popupMsg = new JLabel("The password you entered is invalid.");  
 popupMsg.setBounds(20,10,300,50);  
 popupMsg.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
 popup.add(popupMsg);  
  
 JButton button = new JButton("OK");  
 button.setBounds(120,60,70,20);  
 button.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
 button.addActionListener(actionEvent2 -> {  
 popup.dispose();  
 });  
 popup.add(button);  
  
 popup.setLayout(null);  
 popup.setSize(350, 150);  
 popup.setVisible(true);  
 return false;  
 }  
 }  
 else {  
 JFrame popup = new JFrame("Invalid username");  
 JLabel popupMsg = new JLabel("The username you entered does not exist.");  
 popupMsg.setBounds(20,10,500,50);  
 popupMsg.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
 popup.add(popupMsg);  
  
 JButton button = new JButton("OK");  
 button.setBounds(170,60,70,20);  
 button.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
 button.addActionListener(actionEvent2 -> {  
 popup.dispose();  
 });  
 popup.add(button);  
  
 popup.setLayout(null);  
 popup.setSize(450, 150);  
 popup.setVisible(true);  
 return false;  
 }  
 }

Over here we check if the username exists if it exists we validate the password else we show a popup with a user does not exist message. Similarly, we check for passwords and give a popup if it’s wrong

Invalid user popup

Invalid password popup

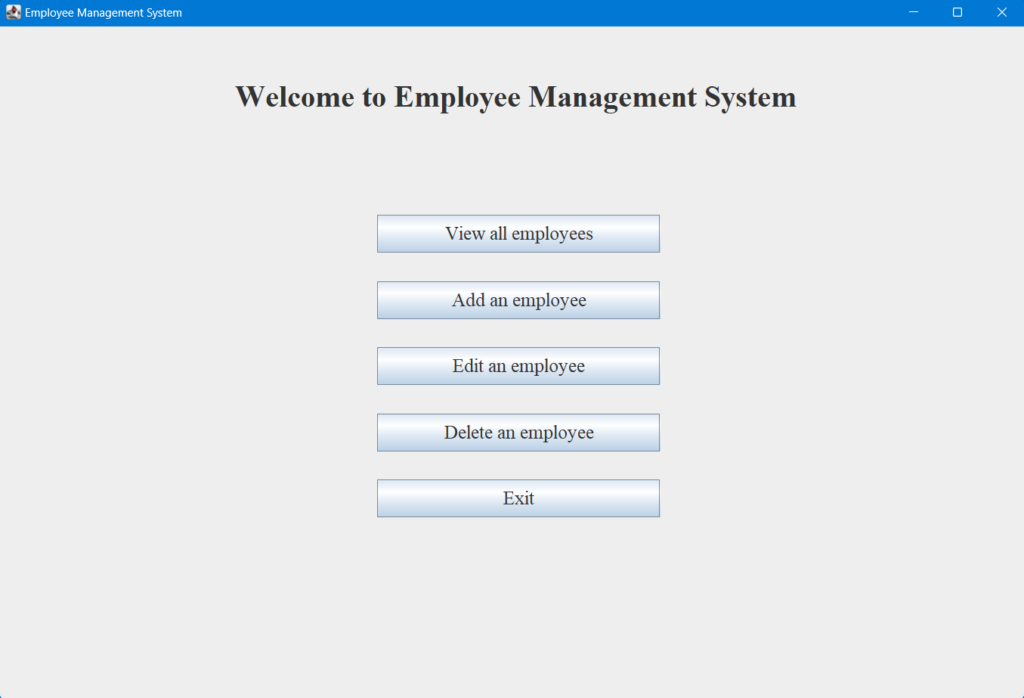
If login is successful we show the main menu

**5. Main Menu in java employee management system**

The main menu will give the user option to view, create, edit and delete employees. Let’s look at how it’s created.

public void mainMenu(){  
 menuFrame = new JFrame("Employee Management System");  
  
 JLabel welcomeLabel = new JLabel("Welcome to Employee Management System");  
 welcomeLabel.setFont(new Font("Times New Roman", Font.*BOLD*, 32));  
 welcomeLabel.setBounds(250,50,800,50);  
 menuFrame.add(welcomeLabel);  
  
 JButton viewEmpButton = new JButton("View all employees");  
 viewEmpButton.setBounds(400, 200, 300, 40);  
 viewEmpButton.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
 viewEmpButton.setFocusPainted(false);  
 viewEmpButton.addActionListener(actionEvent -> {  
 try {  
 viewEmployee();  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
 });  
 menuFrame.add(viewEmpButton);  
  
 JButton addEmpButton = new JButton("Add an employee");  
 addEmpButton.setBounds(400, 270, 300, 40);  
 addEmpButton.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
 addEmpButton.setFocusPainted(false);  
 addEmpButton.addActionListener(actionEvent -> {  
 try {  
 addEmployee();  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
 });  
 menuFrame.add(addEmpButton);  
  
 JButton editEmpButton = new JButton("Edit an employee");  
 editEmpButton.setBounds(400, 340, 300, 40);  
 editEmpButton.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
 editEmpButton.setFocusPainted(false);  
 editEmpButton.addActionListener(actionEvent -> {  
 try {  
 editEmployee();  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
 });  
 menuFrame.add(editEmpButton);  
  
 JButton deleteEmpButton = new JButton("Delete an employee");  
 deleteEmpButton.setBounds(400, 410, 300, 40);  
 deleteEmpButton.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
 deleteEmpButton.setFocusPainted(false);  
 deleteEmpButton.addActionListener(deleteEvent -> {  
 deleteEmployee();  
 });  
 menuFrame.add(deleteEmpButton);  
  
 JButton exitButton = new JButton("Exit");  
 exitButton.setBounds(400, 480, 300, 40);  
 exitButton.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
 exitButton.setFocusPainted(false);  
  
 exitButton.addActionListener(actionEvent -> {  
 menuFrame.dispose();  
 });  
 menuFrame.add(exitButton);  
  
  
 menuFrame.setSize(1100,750);  
 menuFrame.setDefaultCloseOperation(WindowConstants.*EXIT\_ON\_CLOSE*);  
 menuFrame.setLayout(null);//using no layout managers  
 menuFrame.setVisible(true);//making the frame visible  
  
}

We create buttons for each functionality and add action listeners to them that call their respective methods. The login screen will look like this:

Main Menu

Now the user is able to select the options. Let us now discuss how the different functionalities are implemented.

**6. View Screen for viewing employees in employee management project in java**

We will query all employee records from the database and show them. I am using GridLayout to arrange the data easily. We will create a different panel for each employee so that we can have a card-like display.

public void viewEmployee() throws SQLException {

menuFrame.setVisible(false);

JFrame frame=new JFrame("Employee Records");

JPanel panel=new JPanel();

Statement statement = this.con.createStatement();

String q = "select \* from employee";

ResultSet resultSet = statement.executeQuery(q);

while (resultSet.next()) {

JPanel employeeCard = new JPanel();

JLabel idLabel = new JLabel("ID");

idLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel idVal = new JLabel(String.valueOf(resultSet.getInt(1)));

idVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel nameLabel = new JLabel("Name");

nameLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel nameVal = new JLabel(resultSet.getString(2));

nameVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel genderLabel = new JLabel("Gender");

genderLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel genderVal = new JLabel(resultSet.getString(3));

genderVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel phoneLabel = new JLabel("Phone Number");

phoneLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel phoneVal = new JLabel(resultSet.getString(4));

phoneVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel emailLabel = new JLabel("Email");

emailLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel emailVal = new JLabel(resultSet.getString(5));

emailVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel designationLabel = new JLabel("Designation");

designationLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel designationVal = new JLabel(resultSet.getString(6));

designationVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel salaryLabel = new JLabel("Salary");

salaryLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel salaryVal = new JLabel(String.valueOf(resultSet.getDouble(7)));

salaryVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

employeeCard.add(idLabel);

employeeCard.add(idVal);

employeeCard.add(nameLabel);

employeeCard.add(nameVal);

employeeCard.add(genderLabel);

employeeCard.add(genderVal);

employeeCard.add(phoneLabel);

employeeCard.add(phoneVal);

employeeCard.add(emailLabel);

employeeCard.add(emailVal);

employeeCard.add(designationLabel);

employeeCard.add(designationVal);

employeeCard.add(salaryLabel);

employeeCard.add(salaryVal);

employeeCard.setSize(1000, 400);

employeeCard.setBackground(new Color(166, 209, 230));

employeeCard.setBorder(new EmptyBorder(20, 50, 20, 50));

GridLayout cardLayout = new GridLayout(0, 2);

cardLayout.setHgap(5);

cardLayout.setVgap(10);

employeeCard.setLayout(cardLayout);

panel.add(employeeCard);

}

JPanel buttonPanel = new JPanel();

JButton backButton = new JButton("Back");

backButton.setFont(new Font("Times New Roman", Font.PLAIN, 20));

backButton.setBounds(450, 30, 200,50);

backButton.setFocusPainted(false);

backButton.addActionListener(actionListener -> {

frame.dispose();

menuFrame.setVisible(true);

});

buttonPanel.add(backButton);

buttonPanel.setLayout(null);

buttonPanel.setBackground(new Color(254, 251, 246));

panel.add(buttonPanel);

// panel.setPreferredSize(new Dimension(500,500));

GridLayout layout = new GridLayout(0, 1);

layout.setVgap(30);

panel.setLayout(layout);

panel.setBackground(new Color(254, 251, 246));

panel.setBorder(new EmptyBorder(50, 0, 50, 0));

JScrollPane scrollBar=new JScrollPane(panel,JScrollPane.VERTICAL\_SCROLLBAR\_ALWAYS,JScrollPane.HORIZONTAL\_SCROLLBAR\_NEVER);

frame.add(scrollBar);

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

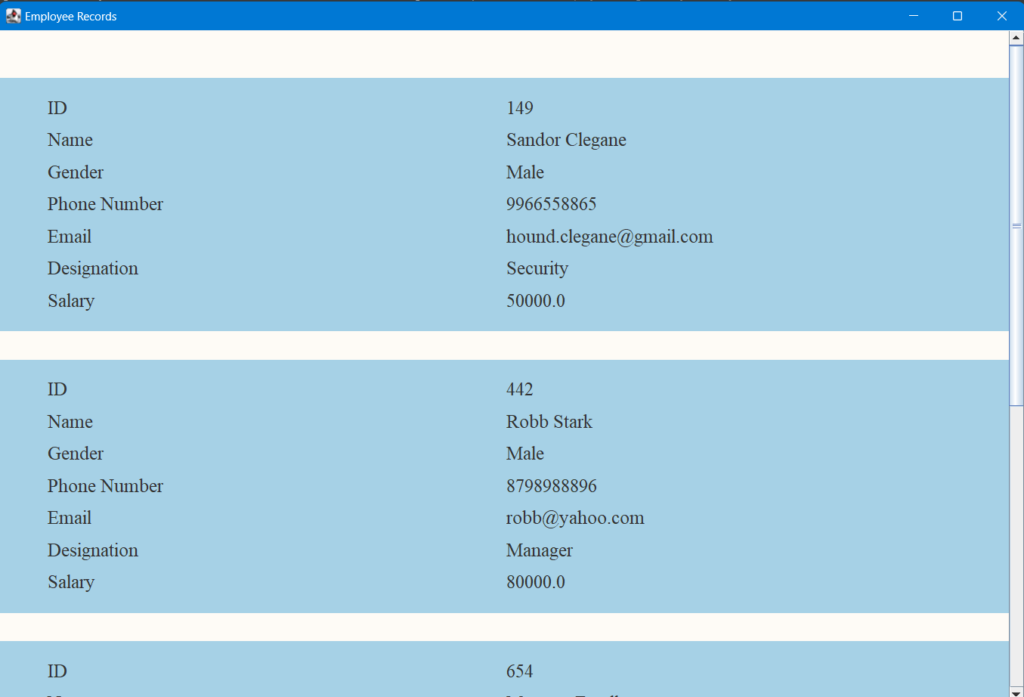
frame.setPreferredSize(new Dimension(1100,750));

frame.pack();

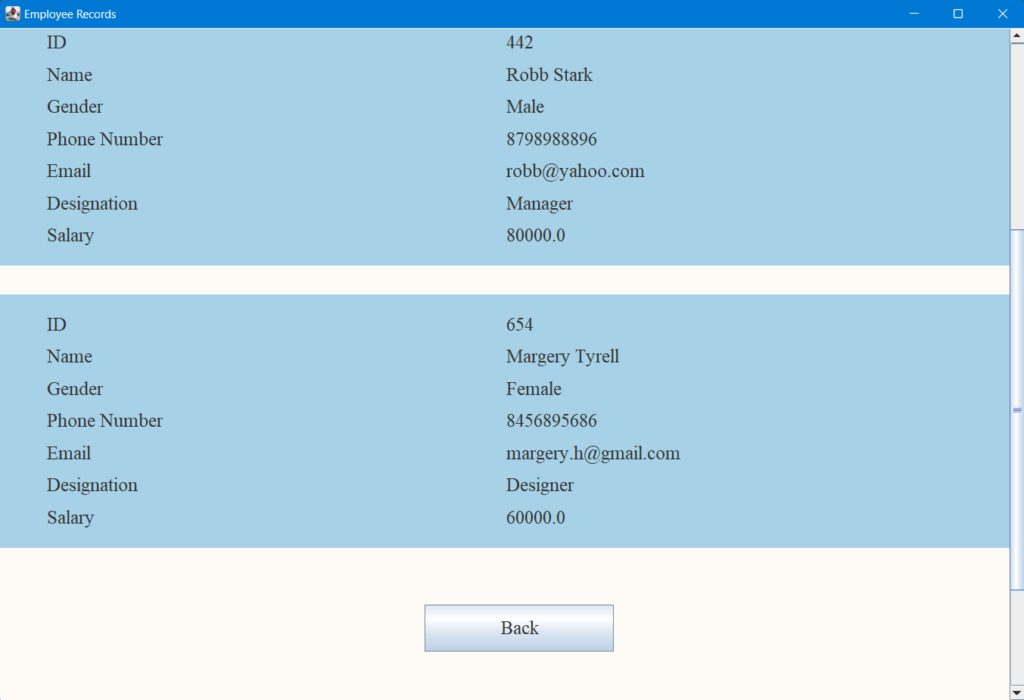
frame.setVisible(true);

}

Let’s view the screen

Employee Records

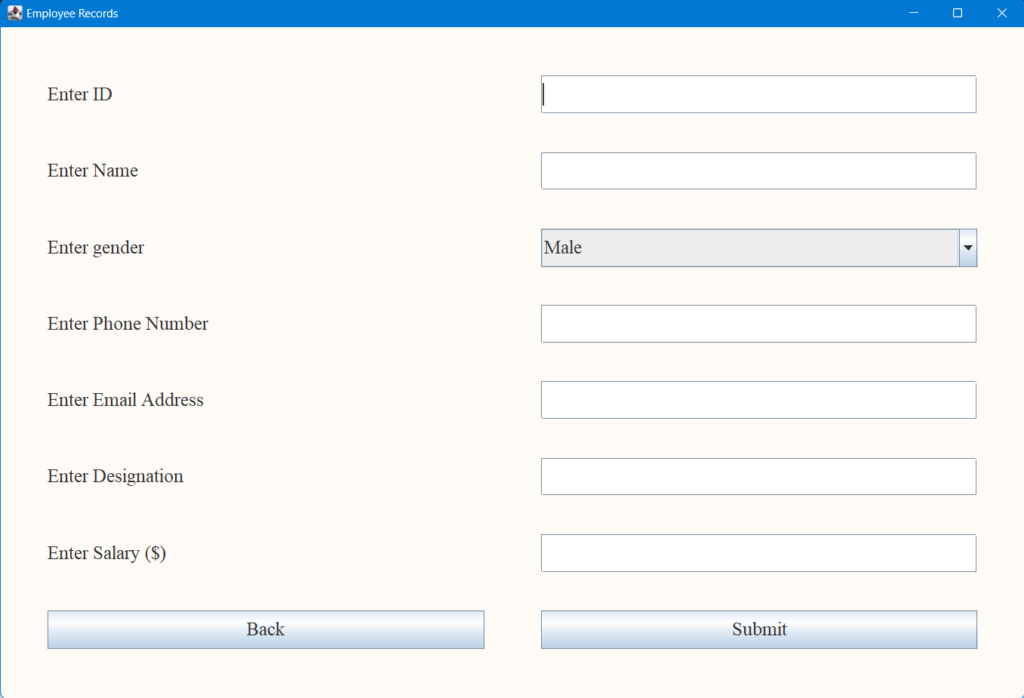
It’s a scrollable interface. You can scroll down to view all data. We have a back button at the end. Pressing it takes us back to the main menu.



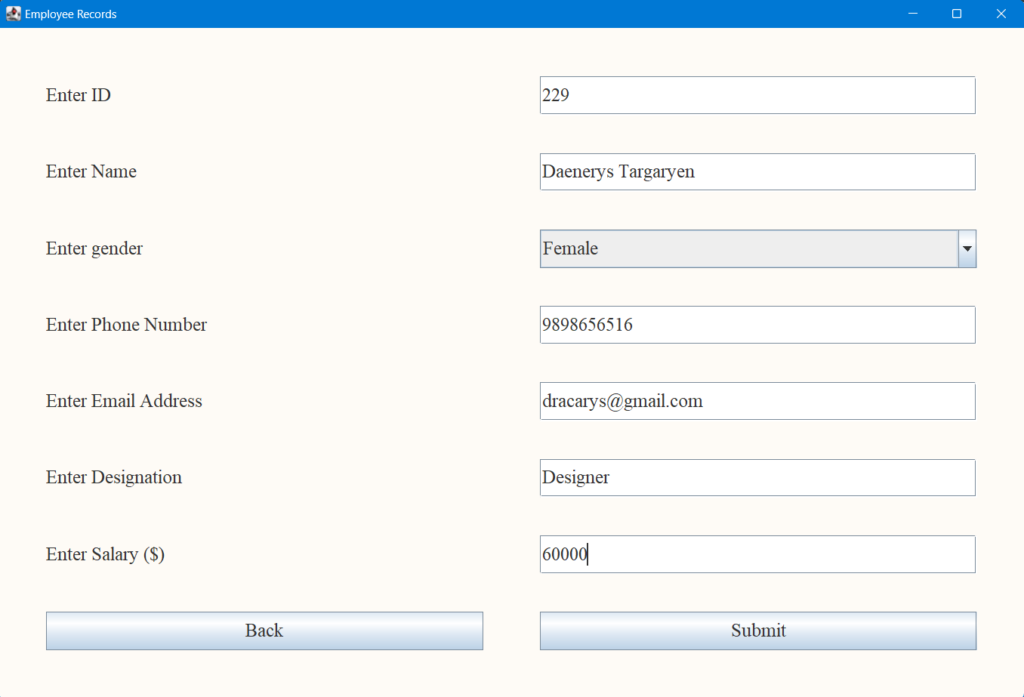
**7. Adding an Employee to the employee management system program**

The add/create employee functionality helps the user to add employees to the database. There are various fields that the user has to fill and click on submit to add the record or click on back to cancel.

public void addEmployee() throws SQLException {  
  
 menuFrame.setVisible(false);  
  
 JFrame frame=new JFrame("Employee Records");  
 JPanel panel=new JPanel();  
  
 JLabel idLabel = new JLabel("Enter ID");  
 idLabel.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JTextField idVal = new JTextField("");  
 idVal.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 idVal.addKeyListener(new KeyAdapter() {  
 public void keyTyped(KeyEvent e) {  
 char c = e.getKeyChar();  
 if ( ((c < '0') || (c > '9')) && (c != KeyEvent.*VK\_BACK\_SPACE*)) {  
 e.consume(); // if it's not a number, ignore the event  
 }  
 }  
 });  
  
 JLabel nameLabel = new JLabel("Enter Name");  
 nameLabel.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JTextField nameVal = new JTextField("");  
 nameVal.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JLabel genderLabel = new JLabel("Enter gender");  
 genderLabel.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JComboBox<String> genderVal = new JComboBox<>(new String[]{"Male", "Female"});  
 genderVal.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JLabel phoneLabel = new JLabel("Enter Phone Number");  
 phoneLabel.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JTextField phoneVal = new JTextField("");  
 phoneVal.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JLabel emailLabel = new JLabel("Enter Email Address");  
 emailLabel.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JTextField emailVal = new JTextField("");  
 emailVal.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JLabel designationLabel = new JLabel("Enter Designation");  
 designationLabel.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JTextField designationVal = new JTextField("");  
 designationVal.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JLabel salaryLabel = new JLabel("Enter Salary ($)");  
 salaryLabel.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JTextField salaryVal = new JTextField("");  
 salaryVal.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 salaryVal.addKeyListener(new KeyAdapter() {  
 public void keyTyped(KeyEvent e) {  
 char c = e.getKeyChar();  
  
 if ( ((c < '0') || (c > '9')) && (c != KeyEvent.*VK\_BACK\_SPACE*)) {  
 e.consume(); // if it's not a number, ignore the event  
 }  
 }  
 });  
  
 panel.add(idLabel);  
 panel.add(idVal);  
 panel.add(nameLabel);  
 panel.add(nameVal);  
 panel.add(genderLabel);  
 panel.add(genderVal);  
 panel.add(phoneLabel);  
 panel.add(phoneVal);  
 panel.add(emailLabel);  
 panel.add(emailVal);  
 panel.add(designationLabel);  
 panel.add(designationVal);  
 panel.add(salaryLabel);  
 panel.add(salaryVal);  
  
  
 GridLayout cardLayout = new GridLayout(0, 2);  
 cardLayout.setHgap(60);  
 cardLayout.setVgap(40);  
 panel.setLayout(cardLayout);  
  
 panel.setSize(1000, 400);  
 panel.setBackground(new Color(166, 209, 230));  
 panel.setBorder(new EmptyBorder(20, 50, 20, 50));  
  
 JButton backButton = new JButton("Back");  
 backButton.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
 backButton.setBounds(450, 30, 200,50);  
 backButton.setFocusPainted(false);  
  
 backButton.addActionListener(actionListener -> {  
 frame.dispose();  
 menuFrame.setVisible(true);  
 });  
 panel.add(backButton);  
  
 JButton submitButton = new JButton("Submit");  
 submitButton.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
 submitButton.setBounds(450, 30, 200,50);  
 submitButton.setFocusPainted(false);  
  
 submitButton.addActionListener(actionListener -> {  
 Statement statement = null;  
 try {  
 statement = this.con.createStatement();  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
  
 int id = Integer.*parseInt*(idVal.getText());  
 String name = nameVal.getText();  
 String gender = (String) genderVal.getSelectedItem();  
 String phoneNum = phoneVal.getText();  
 String email = emailVal.getText();  
 String designation = designationVal.getText();  
 double salary = Double.*parseDouble*(salaryVal.getText());  
  
 String q = String.*format*("insert into employee values (%d, '%s', '%s', '%s', '%s', '%s', %f);", id, name, gender,  
 phoneNum, email, designation, salary);  
 try {  
 statement.executeUpdate(q);  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
  
 menuFrame.setVisible(true);  
 frame.dispose();  
  
 });  
 panel.add(submitButton);  
  
 panel.setBackground(new Color(254, 251, 246));  
 panel.setBorder(new EmptyBorder(50, 50, 50, 50));  
  
 frame.add(panel);  
 frame.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);  
 frame.setPreferredSize(new Dimension(1100,750));  
 frame.pack();  
 frame.setVisible(true);  
  
}

Add Employee

Let’s fill this out and add an employee.



Now lets press submit and check if our record is added.

Go to view all employees and you can see the record.



**8. Editing an Employee**

Edit employee involves editing the different fields of an employee. We start by taking the employee id and checking it is valid.

public void editEmployee() throws SQLException {

menuFrame.setVisible(false);

JFrame frame = new JFrame("Edit Employee");

JLabel label = new JLabel("Enter employee id");

label.setFont(new Font("Times New Roman", Font.*BOLD*, 20));

label.setBounds(250,200,200,50);

frame.add(label);

JTextField idVal = new JTextField();

idVal.setFont(new Font("Times New Roman", Font.*BOLD*, 20));

idVal.setBounds(500,200,200,50);

frame.add(idVal);

idVal.addKeyListener(new KeyAdapter() {

public void keyTyped(KeyEvent e) {

char c = e.getKeyChar();

if ( ((c < '0') || (c > '9')) && (c != KeyEvent.*VK\_BACK\_SPACE*)) {

e.consume(); // if it's not a number, ignore the event

}

}

});

JButton backButton = new JButton("Back");

backButton.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));

backButton.setBounds(275, 400, 150,40);

backButton.setFocusPainted(false);

backButton.addActionListener(actionListener -> {

menuFrame.setVisible(true);

frame.dispose();

});

frame.add(backButton);

JButton submitButton = new JButton("Submit");

submitButton.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));

submitButton.setBounds(525, 400, 150,40);

submitButton.setFocusPainted(false);

submitButton.addActionListener(actionEvent -> {

int id = Integer.*parseInt*(idVal.getText());

String q = String.*format*("select \* from employee where id = %d;", id);

Statement statement = null;

try {

statement = this.con.createStatement();

ResultSet resultSet = statement.executeQuery(q);

if (resultSet.next()) {

System.*out*.println(resultSet.getString(2));

// return true;

editEmployeeHelper(id, frame);

}

else{

JFrame popup = new JFrame("Invalid ID");

JLabel popupMsg = new JLabel("The ID you entered is invalid.");

popupMsg.setBounds(20,10,300,50);

popupMsg.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));

popup.add(popupMsg);

JButton button = new JButton("OK");

button.setBounds(120,60,70,20);

button.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));

button.addActionListener(actionEvent2 -> {

popup.dispose();

});

popup.add(button);

popup.setLayout(null);

popup.setSize(350, 150);

popup.setVisible(true);

}

} catch (SQLException throwables) {

throwables.printStackTrace();

}

});

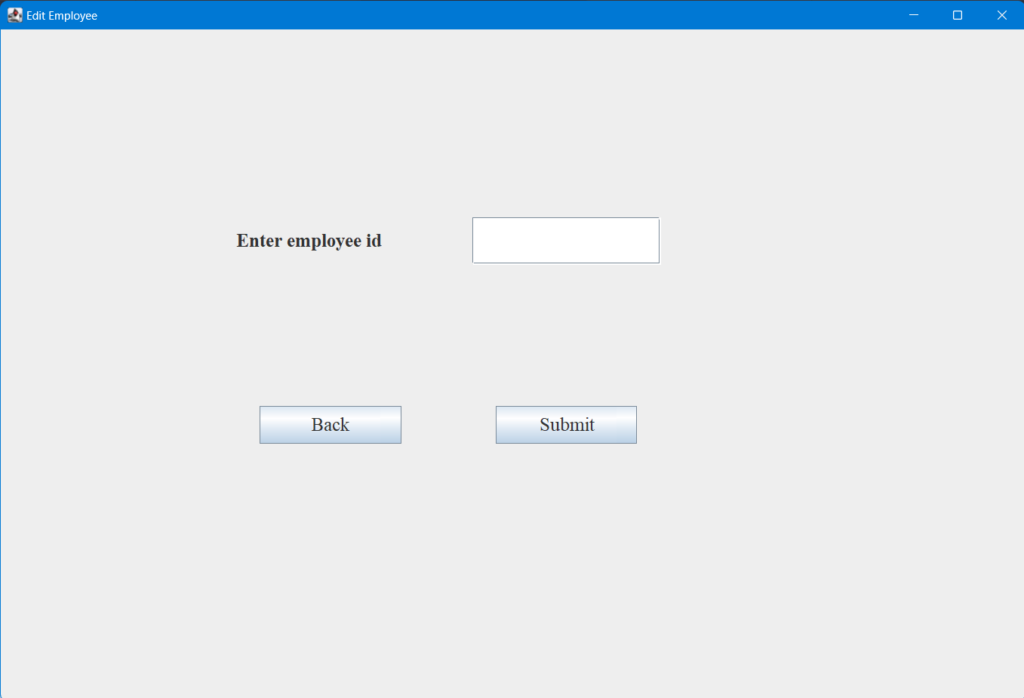
frame.add(submitButton);

frame.setLayout(null);

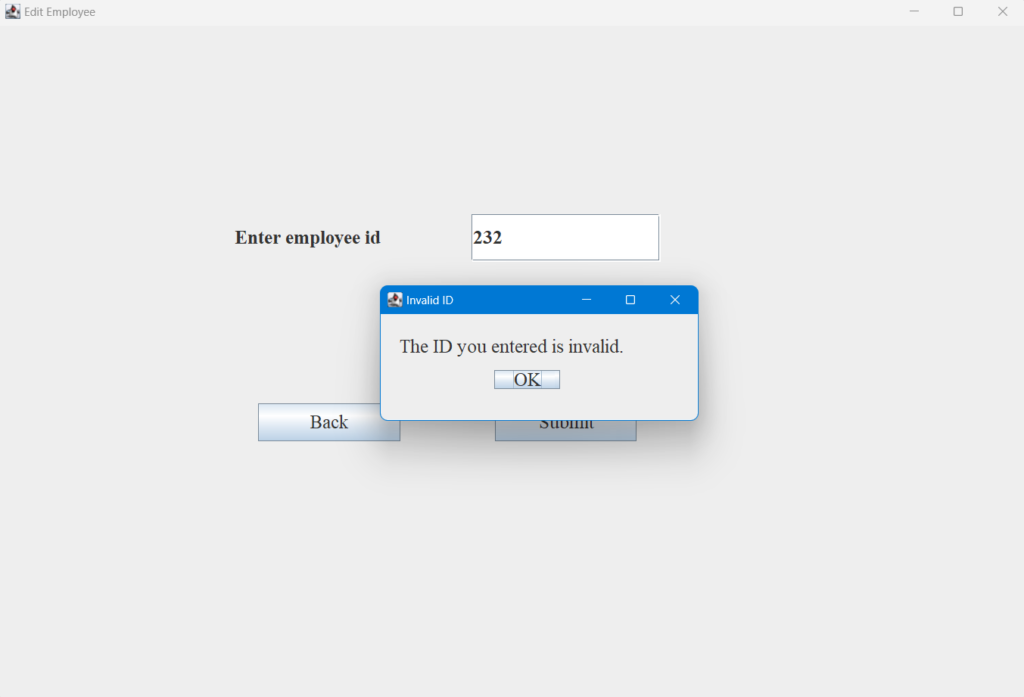
frame.setSize(new Dimension(1100,750));

frame.setVisible(true);

}

Edit employee

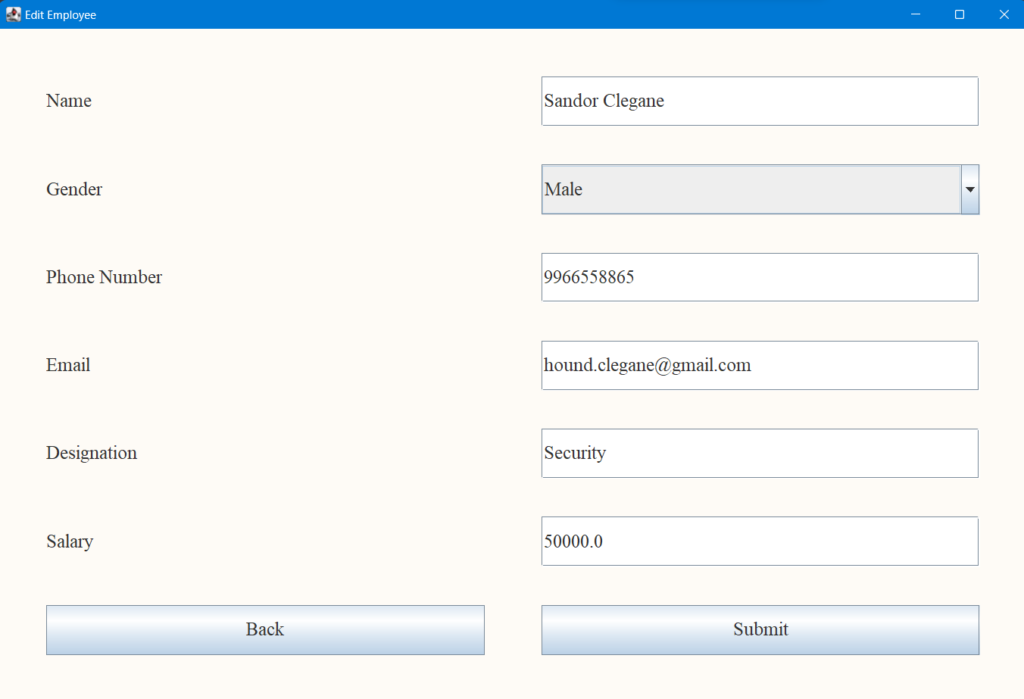
We check the employee id in our DB and if it doesn’t exist we display a popup.



Lets add the edit functionality

public void editEmployeeHelper(int id, JFrame parentFrame) throws SQLException {  
 JFrame frame = new JFrame("Edit Employee");  
 JPanel panel=new JPanel();  
  
 Statement statement = null;  
 try {  
 statement = this.con.createStatement();  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
  
 String q = String.*format*("select \* from employee where id = %d;",id);  
 ResultSet resultSet = statement.executeQuery(q);  
 resultSet.next();  
 JLabel nameLabel = new JLabel("Name");  
 nameLabel.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JTextField nameVal = new JTextField(resultSet.getString(2));  
 nameVal.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JLabel genderLabel = new JLabel("Gender");  
 genderLabel.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 String gender = resultSet.getString(3);  
 JComboBox<String> genderVal = new JComboBox<>(new String[]{"Male", "Female"});  
 genderVal.setSelectedIndex(gender.equals("Male") ? 0 : 1 );  
 genderVal.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JLabel phoneLabel = new JLabel("Phone Number");  
 phoneLabel.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JTextField phoneVal = new JTextField(resultSet.getString(4));  
 phoneVal.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JLabel emailLabel = new JLabel("Email");  
 emailLabel.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JTextField emailVal = new JTextField(resultSet.getString(5));  
 emailVal.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JLabel designationLabel = new JLabel("Designation");  
 designationLabel.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JTextField designationVal = new JTextField(resultSet.getString(6));  
 designationVal.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JLabel salaryLabel = new JLabel("Salary");  
 salaryLabel.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 JTextField salaryVal = new JTextField(String.*valueOf*(resultSet.getDouble(7)));  
 salaryVal.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
  
 salaryVal.addKeyListener(new KeyAdapter() {  
 public void keyTyped(KeyEvent e) {  
 char c = e.getKeyChar();  
  
 if ( ((c < '0') || (c > '9')) && (c != KeyEvent.*VK\_BACK\_SPACE*)) {  
 e.consume(); // if it's not a number, ignore the event  
 }  
 }  
 });  
  
 panel.add(nameLabel);  
 panel.add(nameVal);  
 panel.add(genderLabel);  
 panel.add(genderVal);  
 panel.add(phoneLabel);  
 panel.add(phoneVal);  
 panel.add(emailLabel);  
 panel.add(emailVal);  
 panel.add(designationLabel);  
 panel.add(designationVal);  
 panel.add(salaryLabel);  
 panel.add(salaryVal);  
  
  
 GridLayout cardLayout = new GridLayout(0, 2);  
 cardLayout.setHgap(60);  
 cardLayout.setVgap(40);  
 panel.setLayout(cardLayout);  
  
 panel.setSize(1000, 400);  
 panel.setBackground(new Color(166, 209, 230));  
 panel.setBorder(new EmptyBorder(20, 50, 20, 50));  
  
 JButton backButton = new JButton("Back");  
 backButton.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
 backButton.setBounds(450, 30, 200,50);  
 backButton.setFocusPainted(false);  
  
 backButton.addActionListener(actionListener -> {  
 frame.dispose();  
 menuFrame.setVisible(true);  
 });  
 panel.add(backButton);  
  
 JButton submitButton = new JButton("Submit");  
 submitButton.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
 submitButton.setBounds(450, 30, 200,50);  
 submitButton.setFocusPainted(false);  
  
 submitButton.addActionListener(actionListener -> {  
 Statement statemnt = null;  
 try {  
 statemnt = this.con.createStatement();  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
  
 String name = nameVal.getText();  
 String genderValSelectedItem = (String) genderVal.getSelectedItem();  
 String phoneNum = phoneVal.getText();  
 String email = emailVal.getText();  
 String designation = designationVal.getText();  
 double salary = Double.*parseDouble*(salaryVal.getText());  
  
 String query = String.*format*("update employee set name = '%s', gender = '%s', phoneNum = '%s', email = '%s', " +  
 "designation = '%s', salary = %s where id = %d;", name, genderValSelectedItem,  
 phoneNum, email, designation, salary, id);  
 try {  
 statemnt.executeUpdate(query);  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
  
 menuFrame.setVisible(true);  
 frame.dispose();  
  
 });  
 panel.add(submitButton);  
  
 panel.setBackground(new Color(254, 251, 246));  
 panel.setBorder(new EmptyBorder(50, 50, 50, 50));  
  
 frame.add(panel);  
 frame.setDefaultCloseOperation(JFrame.*EXIT\_ON\_CLOSE*);  
 frame.setPreferredSize(new Dimension(1100,750));  
 frame.pack();  
 frame.setVisible(true);  
 parentFrame.dispose();  
}

If the id entered is valid we get a screen with the details of that employee. And those details are made editable so users can make changes as per their wish



Lets increase the salary to 60000 and check in view employee.

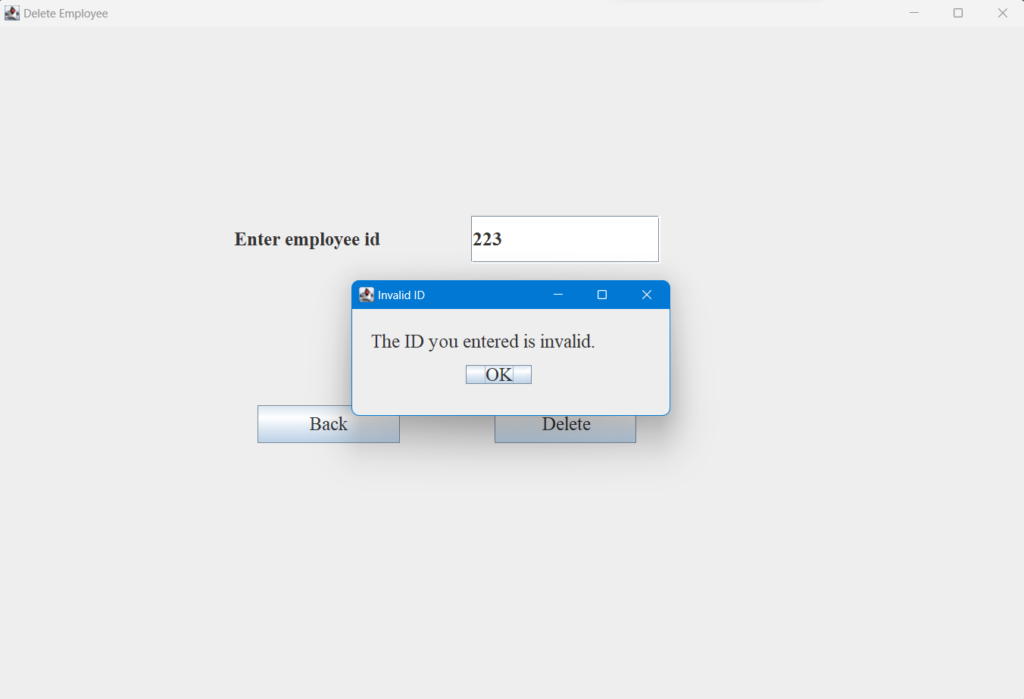


Yes, it gets updated!

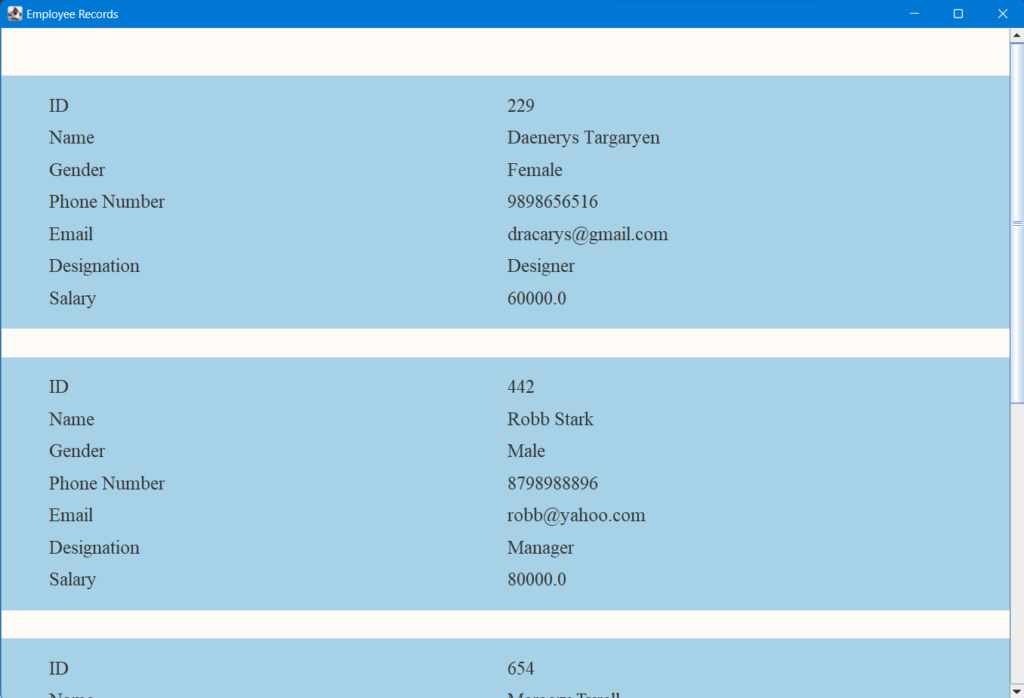
**9. Deleting an Employee in our simple employee management system project in java**

To delete the employee, we again ask for the employee id. If it’s invalid we display a popup else the employee record gets deleted.

public void deleteEmployee(){  
 menuFrame.setVisible(false);  
  
 JFrame frame = new JFrame("Delete Employee");  
  
 JLabel label = new JLabel("Enter employee id");  
 label.setFont(new Font("Times New Roman", Font.*BOLD*, 20));  
 label.setBounds(250,200,200,50);  
 frame.add(label);  
  
 JTextField idVal = new JTextField();  
 idVal.setFont(new Font("Times New Roman", Font.*BOLD*, 20));  
 idVal.setBounds(500,200,200,50);  
 frame.add(idVal);  
 idVal.addKeyListener(new KeyAdapter() {  
 public void keyTyped(KeyEvent e) {  
 char c = e.getKeyChar();  
 if ( ((c < '0') || (c > '9')) && (c != KeyEvent.*VK\_BACK\_SPACE*)) {  
 e.consume(); // if it's not a number, ignore the event  
 }  
 }  
 });  
  
 JButton backButton = new JButton("Back");  
 backButton.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
 backButton.setBounds(275, 400, 150,40);  
 backButton.setFocusPainted(false);  
  
 backButton.addActionListener(actionListener -> {  
 menuFrame.setVisible(true);  
 frame.dispose();  
  
 });  
 frame.add(backButton);  
  
 JButton submitButton = new JButton("Delete");  
 submitButton.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
 submitButton.setBounds(525, 400, 150,40);  
 submitButton.setFocusPainted(false);  
  
 submitButton.addActionListener(actionEvent -> {  
 int id = Integer.*parseInt*(idVal.getText());  
 String q = String.*format*("select \* from employee where id = %d;", id);  
 Statement statement = null;  
 try {  
 statement = this.con.createStatement();  
 ResultSet resultSet = statement.executeQuery(q);  
 if (resultSet.next()) {  
 System.*out*.println(resultSet.getString(2));  
  
 String deleteQuery = String.*format*("delete from employee where id = %d;", id);  
 statement.executeUpdate(deleteQuery);  
 menuFrame.setVisible(true);  
 frame.dispose();  
  
 }  
 else{  
 JFrame popup = new JFrame("Invalid ID");  
 JLabel popupMsg = new JLabel("The ID you entered is invalid.");  
 popupMsg.setBounds(20,10,300,50);  
 popupMsg.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
 popup.add(popupMsg);  
  
 JButton button = new JButton("OK");  
 button.setBounds(120,60,70,20);  
 button.setFont(new Font("Times New Roman", Font.*PLAIN*, 20));  
 button.addActionListener(actionEvent2 -> {  
 popup.dispose();  
 });  
 popup.add(button);  
  
 popup.setLayout(null);  
 popup.setSize(350, 150);  
 popup.setVisible(true);  
  
 }  
 } catch (SQLException throwables) {  
 throwables.printStackTrace();  
 }  
  
 });  
  
 frame.add(submitButton);  
  
 frame.setLayout(null);  
 frame.setSize(new Dimension(1100,750));  
 frame.setVisible(true);  
}



Lets delete our top record that was of Sandor.



Its deleted!

**Complete Source Code for Employee Management System Project in Java**

// get required libraries

import java.awt.\*;

import java.awt.event.KeyAdapter;

import java.awt.event.KeyEvent;

import java.sql.\*;

import java.sql.Connection;

import java.sql.DriverManager;

import java.sql.SQLException;

import javax.swing.\*;

import javax.swing.border.EmptyBorder;

public class EmployeeManagementSystemGUI {

Connection con;

// initialize the system with connection object

EmployeeManagementSystemGUI(Connection con){

this.con = con;

}

public boolean login(String username, String password) throws SQLException {

Statement statement = this.con.createStatement();

String q = String.format("select password from admin where username = '%s';", username);

ResultSet resultSet = statement.executeQuery(q);

// gets us password if the username exists

if(resultSet.next()){

// compare password with user input

if(resultSet.getString(1).equals(password)){

return true;

}

else {

JFrame popup = new JFrame("Invalid password");

JLabel popupMsg = new JLabel("The password you entered is invalid.");

popupMsg.setBounds(20,10,300,50);

popupMsg.setFont(new Font("Times New Roman", Font.PLAIN, 20));

popup.add(popupMsg);

JButton button = new JButton("OK");

button.setBounds(120,60,70,20);

button.setFont(new Font("Times New Roman", Font.PLAIN, 20));

button.addActionListener(actionEvent2 -> {

popup.dispose();

});

popup.add(button);

popup.setLayout(null);

popup.setSize(350, 150);

popup.setVisible(true);

return false;

}

}

else {

JFrame popup = new JFrame("Invalid username");

JLabel popupMsg = new JLabel("The username you entered does not exist.");

popupMsg.setBounds(20,10,500,50);

popupMsg.setFont(new Font("Times New Roman", Font.PLAIN, 20));

popup.add(popupMsg);

JButton button = new JButton("OK");

button.setBounds(170,60,70,20);

button.setFont(new Font("Times New Roman", Font.PLAIN, 20));

button.addActionListener(actionEvent2 -> {

popup.dispose();

});

popup.add(button);

popup.setLayout(null);

popup.setSize(450, 150);

popup.setVisible(true);

return false;

}

}

static String username = "";

static String password = "";

JFrame menuFrame;

JFrame loginFrame;

public void viewEmployee() throws SQLException {

menuFrame.setVisible(false);

JFrame frame=new JFrame("Employee Records");

JPanel panel=new JPanel();

Statement statement = this.con.createStatement();

String q = "select \* from employee";

ResultSet resultSet = statement.executeQuery(q);

while (resultSet.next()) {

JPanel employeeCard = new JPanel();

JLabel idLabel = new JLabel("ID");

idLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel idVal = new JLabel(String.valueOf(resultSet.getInt(1)));

idVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel nameLabel = new JLabel("Name");

nameLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel nameVal = new JLabel(resultSet.getString(2));

nameVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel genderLabel = new JLabel("Gender");

genderLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel genderVal = new JLabel(resultSet.getString(3));

genderVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel phoneLabel = new JLabel("Phone Number");

phoneLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel phoneVal = new JLabel(resultSet.getString(4));

phoneVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel emailLabel = new JLabel("Email");

emailLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel emailVal = new JLabel(resultSet.getString(5));

emailVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel designationLabel = new JLabel("Designation");

designationLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel designationVal = new JLabel(resultSet.getString(6));

designationVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel salaryLabel = new JLabel("Salary");

salaryLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel salaryVal = new JLabel(String.valueOf(resultSet.getDouble(7)));

salaryVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

employeeCard.add(idLabel);

employeeCard.add(idVal);

employeeCard.add(nameLabel);

employeeCard.add(nameVal);

employeeCard.add(genderLabel);

employeeCard.add(genderVal);

employeeCard.add(phoneLabel);

employeeCard.add(phoneVal);

employeeCard.add(emailLabel);

employeeCard.add(emailVal);

employeeCard.add(designationLabel);

employeeCard.add(designationVal);

employeeCard.add(salaryLabel);

employeeCard.add(salaryVal);

employeeCard.setSize(1000, 400);

employeeCard.setBackground(new Color(166, 209, 230));

employeeCard.setBorder(new EmptyBorder(20, 50, 20, 50));

GridLayout cardLayout = new GridLayout(0, 2);

cardLayout.setHgap(5);

cardLayout.setVgap(10);

employeeCard.setLayout(cardLayout);

panel.add(employeeCard);

}

JPanel buttonPanel = new JPanel();

JButton backButton = new JButton("Back");

backButton.setFont(new Font("Times New Roman", Font.PLAIN, 20));

backButton.setBounds(450, 30, 200,50);

backButton.setFocusPainted(false);

backButton.addActionListener(actionListener -> {

frame.dispose();

menuFrame.setVisible(true);

});

buttonPanel.add(backButton);

buttonPanel.setLayout(null);

buttonPanel.setBackground(new Color(254, 251, 246));

panel.add(buttonPanel);

// panel.setPreferredSize(new Dimension(500,500));

GridLayout layout = new GridLayout(0, 1);

layout.setVgap(30);

panel.setLayout(layout);

panel.setBackground(new Color(254, 251, 246));

panel.setBorder(new EmptyBorder(50, 0, 50, 0));

JScrollPane scrollBar=new JScrollPane(panel,JScrollPane.VERTICAL\_SCROLLBAR\_ALWAYS,JScrollPane.HORIZONTAL\_SCROLLBAR\_NEVER);

frame.add(scrollBar);

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setPreferredSize(new Dimension(1100,750));

frame.pack();

frame.setVisible(true);

}

public void addEmployee() throws SQLException {

menuFrame.setVisible(false);

JFrame frame=new JFrame("Employee Records");

JPanel panel=new JPanel();

JLabel idLabel = new JLabel("Enter ID");

idLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JTextField idVal = new JTextField("");

idVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

idVal.addKeyListener(new KeyAdapter() {

public void keyTyped(KeyEvent e) {

char c = e.getKeyChar();

if ( ((c < '0') || (c > '9')) && (c != KeyEvent.VK\_BACK\_SPACE)) {

e.consume(); // if it's not a number, ignore the event

}

}

});

JLabel nameLabel = new JLabel("Enter Name");

nameLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JTextField nameVal = new JTextField("");

nameVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel genderLabel = new JLabel("Enter gender");

genderLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JComboBox<String> genderVal = new JComboBox<>(new String[]{"Male", "Female"});

genderVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel phoneLabel = new JLabel("Enter Phone Number");

phoneLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JTextField phoneVal = new JTextField("");

phoneVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel emailLabel = new JLabel("Enter Email Address");

emailLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JTextField emailVal = new JTextField("");

emailVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel designationLabel = new JLabel("Enter Designation");

designationLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JTextField designationVal = new JTextField("");

designationVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel salaryLabel = new JLabel("Enter Salary ($)");

salaryLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JTextField salaryVal = new JTextField("");

salaryVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

salaryVal.addKeyListener(new KeyAdapter() {

public void keyTyped(KeyEvent e) {

char c = e.getKeyChar();

if ( ((c < '0') || (c > '9')) && (c != KeyEvent.VK\_BACK\_SPACE)) {

e.consume(); // if it's not a number, ignore the event

}

}

});

panel.add(idLabel);

panel.add(idVal);

panel.add(nameLabel);

panel.add(nameVal);

panel.add(genderLabel);

panel.add(genderVal);

panel.add(phoneLabel);

panel.add(phoneVal);

panel.add(emailLabel);

panel.add(emailVal);

panel.add(designationLabel);

panel.add(designationVal);

panel.add(salaryLabel);

panel.add(salaryVal);

GridLayout cardLayout = new GridLayout(0, 2);

cardLayout.setHgap(60);

cardLayout.setVgap(40);

panel.setLayout(cardLayout);

panel.setSize(1000, 400);

panel.setBackground(new Color(166, 209, 230));

panel.setBorder(new EmptyBorder(20, 50, 20, 50));

JButton backButton = new JButton("Back");

backButton.setFont(new Font("Times New Roman", Font.PLAIN, 20));

backButton.setBounds(450, 30, 200,50);

backButton.setFocusPainted(false);

backButton.addActionListener(actionListener -> {

frame.dispose();

menuFrame.setVisible(true);

});

panel.add(backButton);

JButton submitButton = new JButton("Submit");

submitButton.setFont(new Font("Times New Roman", Font.PLAIN, 20));

submitButton.setBounds(450, 30, 200,50);

submitButton.setFocusPainted(false);

submitButton.addActionListener(actionListener -> {

Statement statement = null;

try {

statement = this.con.createStatement();

} catch (SQLException throwables) {

throwables.printStackTrace();

}

int id = Integer.parseInt(idVal.getText());

String name = nameVal.getText();

String gender = (String) genderVal.getSelectedItem();

String phoneNum = phoneVal.getText();

String email = emailVal.getText();

String designation = designationVal.getText();

double salary = Double.parseDouble(salaryVal.getText());

String q = String.format("insert into employee values (%d, '%s', '%s', '%s', '%s', '%s', %f);", id, name, gender,

phoneNum, email, designation, salary);

try {

statement.executeUpdate(q);

} catch (SQLException throwables) {

throwables.printStackTrace();

}

menuFrame.setVisible(true);

frame.dispose();

});

panel.add(submitButton);

panel.setBackground(new Color(254, 251, 246));

panel.setBorder(new EmptyBorder(50, 50, 50, 50));

frame.add(panel);

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setPreferredSize(new Dimension(1100,750));

frame.pack();

frame.setVisible(true);

}

public void editEmployee() throws SQLException {

menuFrame.setVisible(false);

JFrame frame = new JFrame("Edit Employee");

JLabel label = new JLabel("Enter employee id");

label.setFont(new Font("Times New Roman", Font.BOLD, 20));

label.setBounds(250,200,200,50);

frame.add(label);

JTextField idVal = new JTextField();

idVal.setFont(new Font("Times New Roman", Font.BOLD, 20));

idVal.setBounds(500,200,200,50);

frame.add(idVal);

idVal.addKeyListener(new KeyAdapter() {

public void keyTyped(KeyEvent e) {

char c = e.getKeyChar();

if ( ((c < '0') || (c > '9')) && (c != KeyEvent.VK\_BACK\_SPACE)) {

e.consume(); // if it's not a number, ignore the event

}

}

});

JButton backButton = new JButton("Back");

backButton.setFont(new Font("Times New Roman", Font.PLAIN, 20));

backButton.setBounds(275, 400, 150,40);

backButton.setFocusPainted(false);

backButton.addActionListener(actionListener -> {

menuFrame.setVisible(true);

frame.dispose();

});

frame.add(backButton);

JButton submitButton = new JButton("Submit");

submitButton.setFont(new Font("Times New Roman", Font.PLAIN, 20));

submitButton.setBounds(525, 400, 150,40);

submitButton.setFocusPainted(false);

submitButton.addActionListener(actionEvent -> {

int id = Integer.parseInt(idVal.getText());

String q = String.format("select \* from employee where id = %d;", id);

Statement statement = null;

try {

statement = this.con.createStatement();

ResultSet resultSet = statement.executeQuery(q);

if (resultSet.next()) {

System.out.println(resultSet.getString(2));

// return true;

editEmployeeHelper(id, frame);

}

else{

JFrame popup = new JFrame("Invalid ID");

JLabel popupMsg = new JLabel("The ID you entered is invalid.");

popupMsg.setBounds(20,10,300,50);

popupMsg.setFont(new Font("Times New Roman", Font.PLAIN, 20));

popup.add(popupMsg);

JButton button = new JButton("OK");

button.setBounds(120,60,70,20);

button.setFont(new Font("Times New Roman", Font.PLAIN, 20));

button.addActionListener(actionEvent2 -> {

popup.dispose();

});

popup.add(button);

popup.setLayout(null);

popup.setSize(350, 150);

popup.setVisible(true);

}

} catch (SQLException throwables) {

throwables.printStackTrace();

}

});

frame.add(submitButton);

frame.setLayout(null);

frame.setSize(new Dimension(1100,750));

frame.setVisible(true);

}

public void editEmployeeHelper(int id, JFrame parentFrame) throws SQLException {

JFrame frame = new JFrame("Edit Employee");

JPanel panel=new JPanel();

Statement statement = null;

try {

statement = this.con.createStatement();

} catch (SQLException throwables) {

throwables.printStackTrace();

}

String q = String.format("select \* from employee where id = %d;",id);

ResultSet resultSet = statement.executeQuery(q);

resultSet.next();

JLabel nameLabel = new JLabel("Name");

nameLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JTextField nameVal = new JTextField(resultSet.getString(2));

nameVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel genderLabel = new JLabel("Gender");

genderLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

String gender = resultSet.getString(3);

JComboBox<String> genderVal = new JComboBox<>(new String[]{"Male", "Female"});

genderVal.setSelectedIndex(gender.equals("Male") ? 0 : 1 );

genderVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel phoneLabel = new JLabel("Phone Number");

phoneLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JTextField phoneVal = new JTextField(resultSet.getString(4));

phoneVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel emailLabel = new JLabel("Email");

emailLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JTextField emailVal = new JTextField(resultSet.getString(5));

emailVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel designationLabel = new JLabel("Designation");

designationLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JTextField designationVal = new JTextField(resultSet.getString(6));

designationVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JLabel salaryLabel = new JLabel("Salary");

salaryLabel.setFont(new Font("Times New Roman", Font.PLAIN, 20));

JTextField salaryVal = new JTextField(String.valueOf(resultSet.getDouble(7)));

salaryVal.setFont(new Font("Times New Roman", Font.PLAIN, 20));

salaryVal.addKeyListener(new KeyAdapter() {

public void keyTyped(KeyEvent e) {

char c = e.getKeyChar();

if ( ((c < '0') || (c > '9')) && (c != KeyEvent.VK\_BACK\_SPACE)) {

e.consume(); // if it's not a number, ignore the event

}

}

});

panel.add(nameLabel);

panel.add(nameVal);

panel.add(genderLabel);

panel.add(genderVal);

panel.add(phoneLabel);

panel.add(phoneVal);

panel.add(emailLabel);

panel.add(emailVal);

panel.add(designationLabel);

panel.add(designationVal);

panel.add(salaryLabel);

panel.add(salaryVal);

GridLayout cardLayout = new GridLayout(0, 2);

cardLayout.setHgap(60);

cardLayout.setVgap(40);

panel.setLayout(cardLayout);

panel.setSize(1000, 400);

panel.setBackground(new Color(166, 209, 230));

panel.setBorder(new EmptyBorder(20, 50, 20, 50));

JButton backButton = new JButton("Back");

backButton.setFont(new Font("Times New Roman", Font.PLAIN, 20));

backButton.setBounds(450, 30, 200,50);

backButton.setFocusPainted(false);

backButton.addActionListener(actionListener -> {

frame.dispose();

menuFrame.setVisible(true);

});

panel.add(backButton);

JButton submitButton = new JButton("Submit");

submitButton.setFont(new Font("Times New Roman", Font.PLAIN, 20));

submitButton.setBounds(450, 30, 200,50);

submitButton.setFocusPainted(false);

submitButton.addActionListener(actionListener -> {

Statement statemnt = null;

try {

statemnt = this.con.createStatement();

} catch (SQLException throwables) {

throwables.printStackTrace();

}

String name = nameVal.getText();

String genderValSelectedItem = (String) genderVal.getSelectedItem();

String phoneNum = phoneVal.getText();

String email = emailVal.getText();

String designation = designationVal.getText();

double salary = Double.parseDouble(salaryVal.getText());

String query = String.format("update employee set name = '%s', gender = '%s', phoneNum = '%s', email = '%s', " +

"designation = '%s', salary = %s where id = %d;", name, genderValSelectedItem,

phoneNum, email, designation, salary, id);

try {

statemnt.executeUpdate(query);

} catch (SQLException throwables) {

throwables.printStackTrace();

}

menuFrame.setVisible(true);

frame.dispose();

});

panel.add(submitButton);

panel.setBackground(new Color(254, 251, 246));

panel.setBorder(new EmptyBorder(50, 50, 50, 50));

frame.add(panel);

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setPreferredSize(new Dimension(1100,750));

frame.pack();

frame.setVisible(true);

parentFrame.dispose();

}

public void deleteEmployee(){

menuFrame.setVisible(false);

JFrame frame = new JFrame("Delete Employee");

JLabel label = new JLabel("Enter employee id");

label.setFont(new Font("Times New Roman", Font.BOLD, 20));

label.setBounds(250,200,200,50);

frame.add(label);

JTextField idVal = new JTextField();

idVal.setFont(new Font("Times New Roman", Font.BOLD, 20));

idVal.setBounds(500,200,200,50);

frame.add(idVal);

idVal.addKeyListener(new KeyAdapter() {

public void keyTyped(KeyEvent e) {

char c = e.getKeyChar();

if ( ((c < '0') || (c > '9')) && (c != KeyEvent.VK\_BACK\_SPACE)) {

e.consume(); // if it's not a number, ignore the event

}

}

});

JButton backButton = new JButton("Back");

backButton.setFont(new Font("Times New Roman", Font.PLAIN, 20));

backButton.setBounds(275, 400, 150,40);

backButton.setFocusPainted(false);

backButton.addActionListener(actionListener -> {

menuFrame.setVisible(true);

frame.dispose();

});

frame.add(backButton);

JButton submitButton = new JButton("Delete");

submitButton.setFont(new Font("Times New Roman", Font.PLAIN, 20));

submitButton.setBounds(525, 400, 150,40);

submitButton.setFocusPainted(false);

submitButton.addActionListener(actionEvent -> {

int id = Integer.parseInt(idVal.getText());

String q = String.format("select \* from employee where id = %d;", id);

Statement statement = null;

try {

statement = this.con.createStatement();

ResultSet resultSet = statement.executeQuery(q);

if (resultSet.next()) {

System.out.println(resultSet.getString(2));

String deleteQuery = String.format("delete from employee where id = %d;", id);

statement.executeUpdate(deleteQuery);

menuFrame.setVisible(true);

frame.dispose();

}

else{

JFrame popup = new JFrame("Invalid ID");

JLabel popupMsg = new JLabel("The ID you entered is invalid.");

popupMsg.setBounds(20,10,300,50);

popupMsg.setFont(new Font("Times New Roman", Font.PLAIN, 20));

popup.add(popupMsg);

JButton button = new JButton("OK");

button.setBounds(120,60,70,20);

button.setFont(new Font("Times New Roman", Font.PLAIN, 20));

button.addActionListener(actionEvent2 -> {

popup.dispose();

});

popup.add(button);

popup.setLayout(null);

popup.setSize(350, 150);

popup.setVisible(true);

}

} catch (SQLException throwables) {

throwables.printStackTrace();

}

});

frame.add(submitButton);

frame.setLayout(null);

frame.setSize(new Dimension(1100,750));

frame.setVisible(true);

}

public void mainMenu(){

menuFrame = new JFrame("Employee Management System");

JLabel welcomeLabel = new JLabel("Welcome to Employee Management System");

welcomeLabel.setFont(new Font("Times New Roman", Font.BOLD, 32));

welcomeLabel.setBounds(250,50,800,50);

menuFrame.add(welcomeLabel);

JButton viewEmpButton = new JButton("View all employees");

viewEmpButton.setBounds(400, 200, 300, 40);

viewEmpButton.setFont(new Font("Times New Roman", Font.PLAIN, 20));

viewEmpButton.setFocusPainted(false);

viewEmpButton.addActionListener(actionEvent -> {

try {

viewEmployee();

} catch (SQLException throwables) {

throwables.printStackTrace();

}

});

menuFrame.add(viewEmpButton);

JButton addEmpButton = new JButton("Add an employee");

addEmpButton.setBounds(400, 270, 300, 40);

addEmpButton.setFont(new Font("Times New Roman", Font.PLAIN, 20));

addEmpButton.setFocusPainted(false);

addEmpButton.addActionListener(actionEvent -> {

try {

addEmployee();

} catch (SQLException throwables) {

throwables.printStackTrace();

}

});

menuFrame.add(addEmpButton);

JButton editEmpButton = new JButton("Edit an employee");

editEmpButton.setBounds(400, 340, 300, 40);

editEmpButton.setFont(new Font("Times New Roman", Font.PLAIN, 20));

editEmpButton.setFocusPainted(false);

editEmpButton.addActionListener(actionEvent -> {

try {

editEmployee();

} catch (SQLException throwables) {

throwables.printStackTrace();

}

});

menuFrame.add(editEmpButton);

JButton deleteEmpButton = new JButton("Delete an employee");

deleteEmpButton.setBounds(400, 410, 300, 40);

deleteEmpButton.setFont(new Font("Times New Roman", Font.PLAIN, 20));

deleteEmpButton.setFocusPainted(false);

deleteEmpButton.addActionListener(deleteEvent -> {

deleteEmployee();

});

menuFrame.add(deleteEmpButton);

JButton exitButton = new JButton("Exit");

exitButton.setBounds(400, 480, 300, 40);

exitButton.setFont(new Font("Times New Roman", Font.PLAIN, 20));

exitButton.setFocusPainted(false);

exitButton.addActionListener(actionEvent -> {

menuFrame.dispose();

});

menuFrame.add(exitButton);

menuFrame.setSize(1100,750);

menuFrame.setDefaultCloseOperation(WindowConstants.EXIT\_ON\_CLOSE);

menuFrame.setLayout(null);//using no layout managers

menuFrame.setVisible(true);//making the frame visible

}

public static void main(String[] args) throws SQLException, ClassNotFoundException {

// gets the required driver

Class.forName("com.mysql.cj.jdbc.Driver");

// creates connection with sql

String dbName = "ems";

String db\_username = "root";

String db\_password = "master";

Connection con= DriverManager.getConnection(

"jdbc:mysql://localhost:3306/"+dbName, db\_username, db\_password);

EmployeeManagementSystemGUI ems = new EmployeeManagementSystemGUI(con);

ems.loginFrame = new JFrame("Login");

JLabel usernameLabel = new JLabel("Username");

usernameLabel.setBounds(200,150,220,50);

usernameLabel.setFont(new Font("Times New Roman", Font.PLAIN, 30));

ems.loginFrame.add(usernameLabel);

JTextField usernameField = new JTextField();

usernameField.setFont(new Font("Times New Roman", Font.PLAIN, 20));

usernameField.setBounds(450,150,420,50);

ems.loginFrame.add(usernameField);

JLabel passwordLabel = new JLabel("Password");

passwordLabel.setFont(new Font("Times New Roman", Font.PLAIN, 30));

passwordLabel.setBounds(200,250,220,50);

ems.loginFrame.add(passwordLabel);

JPasswordField passwordField = new JPasswordField();

passwordField.setFont(new Font("Times New Roman", Font.PLAIN, 20));

passwordField.setBounds(450,250,420,50);

ems.loginFrame.add(passwordField);

JButton submitButton=new JButton("Submit");

submitButton.setFont(new Font("Times New Roman", Font.PLAIN, 30));

submitButton.setBounds(450,400,250, 50);

submitButton.addActionListener(actionEvent -> {

username = usernameField.getText();

password = String.valueOf(passwordField.getPassword());

try {

boolean auth = ems.login(username, password);

if(auth){

ems.loginFrame.dispose();

ems.mainMenu();

}

} catch (SQLException throwables) {

throwables.printStackTrace();

}

});

ems.loginFrame.add(submitButton);

ems.loginFrame.setSize(1100,750);

ems.loginFrame.setLayout(null);

ems.loginFrame.setVisible(true);

ems.loginFrame.setDefaultCloseOperation(WindowConstants.EXIT\_ON\_CLOSE);

}

}