CSC 285 Session 7

EmployeeManager (MySQL) Part 2

EmployeeDAO

Modifications

// Employee.java

package empmanager;

/\*\*

\* Application name: Employee manager.

\* Author: Yassine Ballaj.

\* Started Date: 3/23/2015.

\* finished Date:

\*/

import javafx.application.Application;

import javafx.collections.FXCollections;

import javafx.collections.ObservableList;

import javafx.event.Event;

import javafx.event.ActionEvent;

import javafx.event.EventHandler;

import java.sql.\*;

import javafx.scene.input.KeyCode;

import javafx.scene.Scene;

import java.util.ArrayList;

import javafx.geometry.Insets;

import javafx.scene.control.Button;

import javafx.scene.image.Image;

import javafx.scene.image.ImageView;

import javafx.scene.layout.HBox;

import javafx.scene.layout.VBox;

import javafx.scene.paint.Color;

import javafx.scene.text.Font;

import javafx.scene.text.Text;

import javafx.stage.Modality;

import javafx.stage.Stage;

/\*\*

\*

\* @author ballaj

\*/

public class EmployeeManager extends Application {

private SearchPane searchPane =new SearchPane();

private Connection conn;

private Statement stm;

private int [] startingPoint=new int []{0};

private ArrayList<Employee> employees =new ArrayList<Employee>();

@Override

public void start(Stage primaryStage) {

/\*\*Connect to a database \*/

connectToDataBase();

listAllEmployee();

employees.addAll(searchPane.data);

/\*\*List all Employee in the database \*/

searchPane.listAllButton.setOnAction(new EventHandler<ActionEvent>() {

public void handle(ActionEvent event) {

searchPane.data.clear();

searchPane.data.addAll(employees);

}

});

/\*\*Add Employee \*/

searchPane.addEmployee.setOnAction(new EventHandler<ActionEvent>(){

public void handle (ActionEvent event)

{

new AddEmployee(primaryStage,stm, searchPane.data, startingPoint,employees);

}

});

/\*\*Delete employee \*/

searchPane.deleteEmployee.setOnAction(e->{

new deleteEmployee(primaryStage,stm,searchPane.data,employees);

});

/\*\*Update Employee \*/

searchPane.updateEmployee.setOnAction(e->{new UpdateEmployee(primaryStage, stm, searchPane.data, startingPoint,employees);});

/\*\*Search \*/

searchPane.setOnKeyPressed(e->{

if(e.getCode()==KeyCode.ENTER)

{

search(primaryStage);

}

});

/\*\*Exit the application\*/

searchPane.exit.setOnAction(new EventHandler<ActionEvent>(){

public void handle(ActionEvent event)

{

System.exit(0);

}

});

/\*\*Create a scene \*/

Scene scene =new Scene(searchPane,610,500);

primaryStage.setScene(scene);

primaryStage.setTitle("Employee Manager");

primaryStage.show();

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String[] args) {

launch(args);

}

public void connectToDataBase()

{

try {

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

conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/EmployeeManager","root","");

stm=conn.createStatement();

}catch(Exception exception)

{

System.out.println(exception.getMessage());

}

}

public void listAllEmployee()

{

try{

String query ="select \* from employee where id>'"+startingPoint[0]+"'";

ResultSet resultSet=stm.executeQuery(query);

while(resultSet.next())

{

searchPane.data.add(new Employee(

Integer.parseInt(resultSet.getString("id")),

resultSet.getString("firstname"),

resultSet.getString("lastname"),

resultSet.getString("middlename"),

resultSet.getString("position"),

resultSet.getDouble("salary")

));

startingPoint[0]=Integer.parseInt(resultSet.getString(1));

}

}catch(Exception exception)

{

System.out.println(exception.getMessage());

}

}

/\*\*Search method \*/

public void search(Stage primaryStage)

{

int index=-1;

int exist=0;

ArrayList<Integer> indexes=new ArrayList<>();

if (searchPane.searchField.getText().equals(""))

{

searchPane.searchField.requestFocus();

}

else

{

if (searchPane.comboBoxSearch.getValue().equals("Id"))

{

for(int i=0; i<employees.size();i++)

{

if (employees.get(i).getId()==Integer.parseInt(searchPane.searchField.getText()))

{

exist=1;

index=i;

}

}

if (exist==1)

{

searchPane.data.clear();

searchPane.data.add(employees.get(index));

startingPoint[0]=0;

}

else

{

popErrorMessage(primaryStage,"Employee Does Not Exist !!!");

}

}

else if (searchPane.comboBoxSearch.getValue().equals("First name"))

{

for(int i=0; i<employees.size();i++)

{

if (employees.get(i).getFirstName().equals(searchPane.searchField.getText()))

{

exist=1;

indexes.add(i);

}

}

if (exist==1)

{

searchPane.data.clear();

for(int i=0; i<indexes.size();i++)

{

searchPane.data.add(employees.get(indexes.get(i)));

}

startingPoint[0]=0;

}

else

{

popErrorMessage(primaryStage,"Employee Does Not Exist!!!");

}

}

else if (searchPane.comboBoxSearch.getValue().equals("Middle name"))

{

for(int i=0; i<employees.size();i++)

{

if (employees.get(i).getMiddleName().equals(searchPane.searchField.getText()))

{

exist=1;

indexes.add(i);

}

}

if (exist==1)

{

searchPane.data.clear();

for(int i=0; i<indexes.size();i++)

{

searchPane.data.add(employees.get(indexes.get(i)));

}

startingPoint[0]=0;

}

else

{

popErrorMessage(primaryStage,"Employee Does Not Exist!!!");

}

}

else if (searchPane.comboBoxSearch.getValue().equals("Last name"))

{

for(int i=0; i<employees.size();i++)

{

if (employees.get(i).getLastName().equals(searchPane.searchField.getText()))

{

exist=1;

indexes.add(i);

}

}

if (exist==1)

{

searchPane.data.clear();

for(int i=0; i<indexes.size();i++)

{

searchPane.data.add(employees.get(indexes.get(i)));

}

startingPoint[0]=0;

}

else

{

popErrorMessage(primaryStage,"Employee Does Not Exist!!!");

}

}

else if (searchPane.comboBoxSearch.getValue().equals("Position"))

{

for(int i=0; i<employees.size();i++)

{

if (employees.get(i).getPosition().equals(searchPane.searchField.getText()))

{

exist=1;

indexes.add(i);

}

}

if (exist==1)

{

searchPane.data.clear();

for(int i=0; i<indexes.size();i++)

{

searchPane.data.add(employees.get(indexes.get(i)));

}

startingPoint[0]=0;

}

else

{

popErrorMessage(primaryStage,"Employee Does Not Exist!!!");

}

}

else if (searchPane.comboBoxSearch.getValue().equals("Salary"))

{

for(int i=0; i<employees.size();i++)

{

if (employees.get(i).getSalary()==Double.parseDouble(searchPane.searchField.getText()))

{

exist=1;

indexes.add(i);

}

}

if (exist==1)

{

searchPane.data.clear();

for(int i=0; i<indexes.size();i++)

{

searchPane.data.add(employees.get(indexes.get(i)));

}

startingPoint[0]=0;

}

else

{

popErrorMessage(primaryStage,"Employee Does Not Exist!!!");

}

}

}

}

/\*\*Error Message \*/

public void popErrorMessage(Stage primaryStage,String text2)

{

/\*\*Create a Stage \*/

final Stage dialog=new Stage ();

dialog.initModality(Modality.APPLICATION\_MODAL);

dialog.initOwner(primaryStage);

/\*\*Set text properties \*/

Text text=new Text();

text.setText(text2);

text.setFill(Color.RED);

text.setFont(Font.font("Times new roman",20));

/\*\*Set image properties \*/

ImageView image =new ImageView(new Image("file:warning.png"));

/\*\*Set the button properties \*/

Button dialogOk=new Button("OK");

dialogOk.setPrefWidth(100);

/\*\*Set dialog HBox properties \*/

HBox dialogHbox=new HBox ();

dialogHbox.setSpacing(10);

dialogHbox.getChildren().addAll(image,text);

HBox.setMargin(image,new Insets(0,0,0,10));

/\*\*Set VBox properties \*/

VBox dialogVBox =new VBox();

dialogVBox.setSpacing(10);

dialogVBox.getChildren().addAll(dialogHbox,dialogOk);

VBox.setMargin(dialogOk,new Insets(0,0,0,170));

/\*\* exit the pop up window\*/

dialogOk.setOnAction(e->{dialog.close();});

/\*\*Create a scene \*/

Scene scene=new Scene (dialogVBox,450,100);

dialog.setScene(scene);

dialog.setTitle("Error Message");

dialog.show();

}

}

// AddEmployee.java

package empmanager;

import javafx.scene.layout.GridPane;

import javafx.scene.layout.VBox;

import javafx.scene.layout.HBox;

import javafx.scene.control.TextField;

import javafx.scene.control.Label;

import javafx.scene.control.Button;

import javafx.scene.input.KeyCode;

import javafx.geometry.Insets;

import javafx.scene.Scene;

import javafx.stage.Stage;

import java.util.ArrayList;

import java.util.Arrays;

import javafx.geometry.Pos;

import javafx.stage.Modality;

import javafx.collections.ObservableList;

import java.sql.\*;

import javafx.event.ActionEvent;

import javafx.event.EventHandler;

import javafx.scene.image.Image;

import javafx.scene.image.ImageView;

import javafx.scene.paint.Color;

import javafx.scene.text.Font;

import javafx.scene.text.Text;

/\*\*

\*

\* @author ballaj

\*/

public class AddEmployee {

/\*\*Create a private data to build the pane \*/

private TextField tfFirstName=new TextField();

private TextField tfMiddleName =new TextField();

private TextField tfLastName=new TextField();

private TextField tfPosition=new TextField();

private TextField tfSalary=new TextField();

private ArrayList<Label> labels= new ArrayList<Label>(Arrays.asList(

new Label("First name: "), new Label("Middle name: "), new Label("Last name: "),

new Label("Position: "), new Label("Salary: ")));

private Button add=new Button("Add");

private Button clear=new Button("Clear");

private Button cancel =new Button ("Cancel");

private GridPane gridPane =new GridPane();

private VBox vbox =new VBox();

private HBox hbox =new HBox();

private Label title =new Label("Add Employee: ");

/\*\*Create a default constructor \*/

public AddEmployee(Stage primaryStage, Statement stm, ObservableList<Employee> data, int [] startingPoint,ArrayList<Employee> employees )

{

title.setStyle("-fx-font-family:calibri;"+"-fx-font-size:17;");

/\*\*Set the text fields properties \*/

tfFirstName.setStyle("-fx-background:yellow;");

tfFirstName.setPrefWidth(200);

tfMiddleName.setStyle("-fx-background:yellow;");

tfLastName.setStyle("-fx-background:yellow;");

tfPosition.setStyle("-fx-background:yellow;");

tfSalary.setStyle("-fx-background:yellow;");

/\*\*Set the label properties \*/

for (int i=0; i<labels.size();i++)

{

labels.get(i).setStyle("-fx-text-fill:black;"+"-fx-font-family:calibri;"+"-fx-font-size:15;");

}

/\*\*Set the button properties\*/

add.setPrefWidth(100);

clear.setPrefWidth(100);

cancel.setPrefWidth(100);

/\*\*Set the grid pane properties \*/

gridPane.setHgap(10);

gridPane.setVgap(10);

gridPane.setAlignment(Pos.CENTER);

gridPane.add(labels.get(0),0,0);

gridPane.add(tfFirstName,1,0);

gridPane.add(labels.get(1),0,1);

gridPane.add(tfMiddleName,1,1);

gridPane.add(labels.get(2),0,2);

gridPane.add(tfLastName,1,2);

gridPane.add(labels.get(3),0,3);

gridPane.add(tfPosition,1,3);

gridPane.add(labels.get(4), 0, 4);

gridPane.add(tfSalary,1,4);

gridPane.add(hbox,0,5);

/\*\*Set the HBox properties \*/

hbox.setAlignment(Pos.CENTER);

hbox.setSpacing(20);

hbox.getChildren().addAll(add,clear,cancel);

/\*\*Set the VBox properties \*/

vbox.setSpacing(5);

vbox.setPadding(new Insets(10,15,0,0));

vbox.getChildren().addAll(title,gridPane,hbox);

/\*\* Create a Scene \*/

Scene scene=new Scene(vbox,400,300);

final Stage mainStage=new Stage();

mainStage.initModality(Modality.APPLICATION\_MODAL);

mainStage.initOwner(primaryStage);

mainStage.setScene(scene);

mainStage.setTitle("Add Employee");

mainStage.show();

/\*\*Register and handle the event fired by the clear button \*/

clear.setOnAction(e->{clearTextFields();});

/\*\*Register and handle the event fires by the exit button \*/

cancel.setOnAction(e->{mainStage.close();});

/\*\*Register and handle \*/

add.setOnAction(e->{

try{

addNewEmployee(primaryStage,stm,data, startingPoint,employees);

}catch(Exception exception)

{

popErrorMessage(mainStage,exception.getMessage());

}

});

/\*\*Enter Key \*/

vbox.setOnKeyPressed(e->{

if(e.getCode()==KeyCode.ENTER)

{

try{

addNewEmployee(primaryStage,stm,data, startingPoint,employees);

}catch(Exception exception)

{

popErrorMessage(mainStage,exception.getMessage());

}

}

});

}

/\*\*Clear text fields \*/

public void clearTextFields ()

{

tfFirstName.clear();

tfMiddleName.clear();

tfLastName.clear();

tfPosition.clear();

tfSalary.clear();

}

/\*\*Add new employee \*/

public void addNewEmployee(Stage stage,Statement stm, ObservableList<Employee> data, int [] startingPoint, ArrayList<Employee> employees) throws Exception

{

if (tfFirstName.getText().equals(""))

{

tfFirstName.requestFocus();

}

else if (tfMiddleName.getText().equals(""))

{

tfMiddleName.requestFocus();

}

else if (tfLastName.getText().equals(""))

{

tfLastName.requestFocus();

}

else if(tfPosition.getText().equals(""))

{

tfPosition.requestFocus();

}

else if (tfSalary.getText().equals(""))

{

tfSalary.requestFocus();

}

else {

String query ="insert into employee (firstname, middlename, lastname, position, salary) values "

+ "('"+tfFirstName.getText()+"','"+tfMiddleName.getText()+"','"+tfLastName.getText()+"','"+tfPosition.getText()+"','"+tfSalary.getText()+"')";

stm.executeUpdate(query);

clearTextFields();

infoDialog(stage,"Employee Was Added Successfully!!!");

query ="select \* from employee where id>'"+startingPoint[0]+"'";

ResultSet resultSet=stm.executeQuery(query);

while(resultSet.next())

{

data.add(new Employee(

Integer.parseInt(resultSet.getString("id")),

resultSet.getString("firstname"),

resultSet.getString("lastname"),

resultSet.getString("middlename"),

resultSet.getString("position"),

resultSet.getDouble("salary")

));

employees.add(new Employee(

Integer.parseInt(resultSet.getString("id")),

resultSet.getString("firstname"),

resultSet.getString("lastname"),

resultSet.getString("middlename"),

resultSet.getString("position"),

resultSet.getDouble("salary")

));

startingPoint[0]=Integer.parseInt(resultSet.getString(1));

}

}

}

/\*\*Create Information Dialog method \*/

public void infoDialog (Stage primaryStage, String text1)

{

/\*\*Create a Stage \*/

final Stage dialog=new Stage ();

dialog.initModality(Modality.APPLICATION\_MODAL);

dialog.initOwner(primaryStage);

/\*\*Set text properties \*/

Text localText=new Text();

localText.setText(text1);

localText.setFill(Color.BLUE);

localText.setFont(Font.font("Times new roman",20));

/\*\*Set image properties \*/

ImageView image =new ImageView(new Image("file:info.jpg"));

/\*\*Set dialog HBox properties \*/

HBox dialogHbox=new HBox();

dialogHbox.setSpacing(10);

dialogHbox.getChildren().addAll(image,localText);

HBox.setMargin(image,new Insets(0,0,0,10));

/\*\*Set VBox properties \*/

VBox dialogVBox=new VBox();

Button dialogOk=new Button("Ok");

dialogVBox.setSpacing(10);

dialogVBox.getChildren().addAll(dialogHbox,dialogOk);

VBox.setMargin(dialogOk,new Insets(0,0,0,170));

/\*\*Set the button properties \*/

dialogOk.setPrefWidth(100);

/\*\* exit the pop up window\*/

dialogOk.setOnAction(new EventHandler <ActionEvent>(){

public void handle (ActionEvent event)

{

dialog.close();

}

});

/\*\*Create a scene \*/

Scene scene=new Scene (dialogVBox,450,100);

dialog.setScene(scene);

dialog.setTitle("Info Message");

dialog.show();

}

/\*\*Error Message \*/

public void popErrorMessage(Stage primaryStage,String text2)

{

/\*\*Create a Stage \*/

final Stage dialog=new Stage ();

dialog.initModality(Modality.APPLICATION\_MODAL);

dialog.initOwner(primaryStage);

/\*\*Set text properties \*/

Text text=new Text();

text.setText(text2);

text.setFill(Color.RED);

text.setFont(Font.font("Times new roman",20));

/\*\*Set image properties \*/

ImageView image =new ImageView(new Image("file:warning.png"));

/\*\*Set the button properties \*/

Button dialogOk=new Button("OK");

dialogOk.setPrefWidth(100);

/\*\*Set dialog HBox properties \*/

HBox dialogHbox=new HBox ();

dialogHbox.setSpacing(10);

dialogHbox.getChildren().addAll(image,text);

HBox.setMargin(image,new Insets(0,0,0,10));

/\*\*Set VBox properties \*/

VBox dialogVBox =new VBox();

dialogVBox.setSpacing(10);

dialogVBox.getChildren().addAll(dialogHbox,dialogOk);

VBox.setMargin(dialogOk,new Insets(0,0,0,170));

/\*\* exit the pop up window\*/

dialogOk.setOnAction(e->{dialog.close();});

/\*\*Create a scene \*/

Scene scene=new Scene (dialogVBox,450,100);

dialog.setScene(scene);

dialog.setTitle("Error Message");

dialog.show();

}

}

//UpdateEmployee.java

package empmanager;

import java.sql.Statement;

import java.util.ArrayList;

import java.util.Arrays;

import javafx.collections.ObservableList;

import javafx.event.ActionEvent;

import javafx.event.EventHandler;

import javafx.scene.layout.GridPane;

import javafx.scene.layout.HBox;

import javafx.scene.control.TextField;

import javafx.scene.control.Button;

import javafx.scene.control.Label;

import javafx.scene.text.Text;

import javafx.geometry.Pos;

import javafx.geometry.Insets;

import javafx.scene.Scene;

import javafx.scene.image.Image;

import javafx.scene.image.ImageView;

import javafx.scene.input.KeyCode;

import javafx.scene.layout.VBox;

import javafx.scene.paint.Color;

import javafx.scene.text.Font;

import javafx.stage.Modality;

import javafx.stage.Stage;

public class UpdateEmployee {

int elementPosition=-1;

/\*\*Create text fields \*/

private TextField tfId=new TextField();

private TextField tfFirstName=new TextField ();

private TextField tfMiddleName =new TextField();

private TextField tfLastName =new TextField();

private TextField tfPosition =new TextField();

private TextField tfSalary =new TextField ();

/\*\*Create Labels \*/

private ArrayList<Label> labels= new ArrayList<Label>(Arrays.asList( new Label("Id"),

new Label("First name: "), new Label("Middle name: "), new Label("Last name: "),

new Label("Position: "), new Label("Salary: ")));

private Button update=new Button("Update");

private Button clear=new Button("Clear");

private Button cancel =new Button ("Cancel");

private GridPane gridPane =new GridPane();

private VBox vbox =new VBox();

private HBox hbox =new HBox();

private Label title =new Label("Update employee: ");

/\*\*Create a default constructor \*/

public UpdateEmployee (Stage primaryStage, Statement stm, ObservableList<Employee> data, int [] startingPoint, ArrayList<Employee> employees )

{

title.setStyle("-fx-font-family:calibri;"+"-fx-font-size:17;");

/\*\*Set the text fields properties \*/

tfFirstName.setStyle("-fx-background:yellow;");

tfFirstName.setPrefWidth(200);

tfMiddleName.setStyle("-fx-background:yellow;");

tfLastName.setStyle("-fx-background:yellow;");

tfPosition.setStyle("-fx-background:yellow;");

tfSalary.setStyle("-fx-background:yellow;");

/\*\*Set the label properties \*/

for (int i=0; i<labels.size();i++)

{

labels.get(i).setStyle("-fx-text-fill:black;"+"-fx-font-family:calibri;"+"-fx-font-size:15;");

}

/\*\*Set the button properties\*/

update.setPrefWidth(100);

clear.setPrefWidth(100);

cancel.setPrefWidth(100);

/\*\*Set the grid pane properties \*/

gridPane.setHgap(10);

gridPane.setVgap(10);

gridPane.setAlignment(Pos.CENTER);

gridPane.add(labels.get(0),0,0);

gridPane.add(tfId, 1, 0);

gridPane.add(labels.get(1),0,1);

gridPane.add(tfFirstName,1,1);

gridPane.add(labels.get(2),0,2);

gridPane.add(tfMiddleName,1,2);

gridPane.add(labels.get(3),0,3);

gridPane.add(tfLastName,1,3);

gridPane.add(labels.get(4),0,4);

gridPane.add(tfPosition,1,4);

gridPane.add(labels.get(5), 0,5);

gridPane.add(tfSalary,1,5);

gridPane.add(hbox,0,6);

/\*\*Set the HBox properties \*/

hbox.setAlignment(Pos.CENTER);

hbox.setSpacing(20);

hbox.getChildren().addAll(update,clear,cancel);

/\*\*Set the VBox properties \*/

vbox.setSpacing(5);

vbox.setPadding(new Insets(10,15,0,0));

vbox.getChildren().addAll(title,gridPane,hbox);

/\*\* Create a Scene \*/

Scene scene=new Scene(vbox,400,300);

final Stage mainStage=new Stage();

mainStage.initModality(Modality.APPLICATION\_MODAL);

mainStage.initOwner(primaryStage);

mainStage.setScene(scene);

mainStage.setTitle("Add Employee");

mainStage.show();

/\*\*Register and handle the event fired by the clear button \*/

clear.setOnAction(e->{clearTextFields();});

/\*\*Register and handle the event fires by the exit button \*/

cancel.setOnAction(e->{mainStage.close();});

/\*\*Register and handle \*/

update.setOnAction(e->{

try{

updateEmployee(mainStage,data,stm,employees);

}catch(Exception exception)

{

popErrorMessage(mainStage,exception.getMessage());

}

});

/\*\*Enter Key

vbox.setOnKeyPressed(e->{

if(e.getCode()==KeyCode.ENTER)

{

try{

updateEmployee(mainStage,data,stm,employees);

}catch(Exception exception)

{

popErrorMessage(mainStage,exception.getMessage());

}

}

});\*/

/\*\* Employee information \*/

tfId.setOnKeyPressed(e->{

if(e.getCode()==KeyCode.ENTER)

{

try{

employeeInfo(data,mainStage);

}catch(Exception exception)

{

popErrorMessage(mainStage,exception.getMessage());

}

}

});

}

/\*\*Clear text fields \*/

public void clearTextFields ()

{

tfId.clear ();

tfFirstName.clear();

tfMiddleName.clear();

tfLastName.clear();

tfPosition.clear();

tfSalary.clear();

}

/\*\*Create Information Dialog method \*/

public void infoDialog (Stage primaryStage, String text1)

{

/\*\*Create a Stage \*/

final Stage dialog=new Stage ();

dialog.initModality(Modality.APPLICATION\_MODAL);

dialog.initOwner(primaryStage);

/\*\*Set text properties \*/

Text localText=new Text();

localText.setText(text1);

localText.setFill(Color.BLUE);

localText.setFont(Font.font("Times new roman",20));

/\*\*Set image properties \*/

ImageView image =new ImageView(new Image("file:info.jpg"));

/\*\*Set dialog HBox properties \*/

HBox dialogHbox=new HBox();

dialogHbox.setSpacing(10);

dialogHbox.getChildren().addAll(image,localText);

HBox.setMargin(image,new Insets(0,0,0,10));

/\*\*Set VBox properties \*/

VBox dialogVBox=new VBox();

Button dialogOk=new Button("Ok");

dialogVBox.setSpacing(10);

dialogVBox.getChildren().addAll(dialogHbox,dialogOk);

VBox.setMargin(dialogOk,new Insets(0,0,0,170));

/\*\*Set the button properties \*/

dialogOk.setPrefWidth(100);

/\*\* exit the pop up window\*/

dialogOk.setOnAction(new EventHandler <ActionEvent>(){

public void handle (ActionEvent event)

{

dialog.close();

}

});

/\*\*Create a scene \*/

Scene scene=new Scene (dialogVBox,450,100);

dialog.setScene(scene);

dialog.setTitle("Info Message");

dialog.show();

}

/\*\*Error Message \*/

public void popErrorMessage(Stage primaryStage,String text2)

{

/\*\*Create a Stage \*/

final Stage dialog=new Stage ();

dialog.initModality(Modality.APPLICATION\_MODAL);

dialog.initOwner(primaryStage);

/\*\*Set text properties \*/

Text text=new Text();

text.setText(text2);

text.setFill(Color.RED);

text.setFont(Font.font("Times new roman",20));

/\*\*Set image properties \*/

ImageView image =new ImageView(new Image("file:warning.png"));

/\*\*Set the button properties \*/

Button dialogOk=new Button("OK");

dialogOk.setPrefWidth(100);

/\*\*Set dialog HBox properties \*/

HBox dialogHbox=new HBox ();

dialogHbox.setSpacing(10);

dialogHbox.getChildren().addAll(image,text);

HBox.setMargin(image,new Insets(0,0,0,10));

/\*\*Set VBox properties \*/

VBox dialogVBox =new VBox();

dialogVBox.setSpacing(10);

dialogVBox.getChildren().addAll(dialogHbox,dialogOk);

VBox.setMargin(dialogOk,new Insets(0,0,0,170));

/\*\* exit the pop up window\*/

dialogOk.setOnAction(e->{dialog.close();});

/\*\*Create a scene \*/

Scene scene=new Scene (dialogVBox,450,100);

dialog.setScene(scene);

dialog.setTitle("Error Message");

dialog.show();

}

/\*\*Update method \*/

public void employeeInfo(ObservableList<Employee> data, Stage stage) throws Exception

{

int exist=0;

if(tfId.getText().equals(""))

{

tfId.requestFocus();

}

else

{

for(int i=0;i<data.size();i++)

{

if(data.get(i).getId()==Integer.parseInt(tfId.getText()))

{

exist=1;

elementPosition=i;

break;

}

}

if(exist==1)

{

tfFirstName.setText(data.get(elementPosition).getFirstName());

tfMiddleName.setText(data.get(elementPosition).getMiddleName());

tfLastName.setText(data.get(elementPosition).getLastName());

tfPosition.setText(data.get(elementPosition).getPosition());

tfSalary.setText(String.valueOf(data.get(elementPosition).getSalary()));

}

else

{

popErrorMessage(stage,"Employee does not exist.\nPlease Enter Another Id Number!!!");

}

}

}

/\*\*Update Employee \*/

public void updateEmployee(Stage stage, ObservableList<Employee> data, Statement stm, ArrayList<Employee> employees) throws Exception

{

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

String firstName=tfFirstName.getText();

String middleName=tfMiddleName.getText();

String lastName=tfLastName.getText();

String position =tfPosition.getText();

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

String query="update employee set firstname='"+firstName+"', middlename='"+middleName+"', lastname='"+lastName+"', position='"+position+"', salary='"+salary+"' where id='"+id+"'";

if(tfSalary.getText().equals(""))

{

tfId.requestFocus();

}

else

{

stm.executeUpdate(query);

data.set(elementPosition, new Employee(id,firstName, lastName, middleName, position, salary));

employees.set(elementPosition, new Employee(id,firstName, lastName, middleName, position, salary));

clearTextFields();

infoDialog(stage,"Employee was successfully updated.");

}

}

}

// DeleteEmployee.java

package empmanager;

import javafx.scene.paint.Color;

import javafx.scene.layout.GridPane;

import javafx.scene.layout.VBox;

import javafx.scene.layout.HBox;

import javafx.scene.control.TextField;

import javafx.scene.control.Label;

import javafx.scene.control.Button;

import javafx.geometry.Pos;

import javafx.collections.ObservableList;

import javafx.scene.Scene;

import javafx.geometry.Insets;

import javafx.geometry.HPos;

import javafx.stage.Modality;

import javafx.stage.Stage;

import java.sql.\*;

import java.util.ArrayList;

import javafx.event.ActionEvent;

import javafx.event.EventHandler;

import javafx.scene.text.Text;

import javafx.scene.image.ImageView;

import javafx.scene.image.Image;

import javafx.scene.text.Font;

import javafx.scene.input.KeyEvent;

import javafx.scene.input.KeyCode;

import javax.swing.\*;

/\*\*

\*

\* @author ballaj

\*/

public class deleteEmployee {

/\*\*Create a private data \*/

private TextField tfId=new TextField();

private Label idLabel=new Label("Id");

private Label titleLabel=new Label("Delete employee: ");

private Button delete =new Button ("Delete");

private Button cancel =new Button("Cancel");

private VBox vbox=new VBox();

private HBox hbox=new HBox();

private GridPane gridPane =new GridPane();

public deleteEmployee(Stage primaryStage, Statement stm, ObservableList<Employee> data, ArrayList<Employee> employees)

{

tfId.setStyle("-fx-background:yellow;");

tfId.setPrefWidth(200);

idLabel.setStyle("-fx-text-fill: black;"+"-fx-font-family:calibri;"+"-fx-font-size:14;");

titleLabel.setStyle("-fx-text-fill: black;"+"-fx-font-family:calibri;"+"-fx-font-size:16;");

delete.setPrefWidth(100);

cancel.setPrefWidth(100);

hbox.setSpacing(10);

hbox.getChildren().addAll(delete, cancel);

gridPane.setAlignment(Pos.CENTER);

gridPane.setHgap(15);

gridPane.setVgap(20);

gridPane.setPadding (new Insets(10,20,0,0));

gridPane.add(idLabel,0,0);

gridPane.add(tfId,1,0);

gridPane.add(hbox,1,1);

GridPane.setHalignment(hbox,HPos.RIGHT);

vbox.setSpacing(25);

vbox.setPadding(new Insets(10,20,0,0));

vbox.getChildren().addAll(titleLabel,gridPane);

Scene scene =new Scene(vbox,300,150);

Stage mainStage=new Stage();

mainStage.initModality(Modality.APPLICATION\_MODAL);

mainStage.initOwner(primaryStage);

mainStage.setScene(scene);

mainStage.setTitle("Employee Manager App (Delete employee)");

mainStage.show();

/\*\*Register and handle the event fired by cancel method \*/

cancel.setOnAction(e->{mainStage.close();});

/\*\*Register and handle the event fired by the cancel method \*/

delete.setOnAction(e->{

try{

deleteEmployee(mainStage,stm,data,employees);

}catch (Exception exception)

{

popErrorMessage(mainStage, exception.getMessage());

}

});

/\*\*Register and handle the event fired by the Enter key \*/

/\*\*Enter Key \*/

vbox.setOnKeyPressed(e->{

if(e.getCode()==KeyCode.ENTER)

{

try{

deleteEmployee(primaryStage, stm, data,employees);

}catch(Exception exception)

{

popErrorMessage(mainStage,exception.getMessage());

}

}

});

}

/\*\*Delete employee \*/

public void deleteEmployee(Stage stage, Statement stm, ObservableList<Employee> data, ArrayList<Employee> employees)throws Exception

{

int exist=0;

String query ="delete from employee where id='"+Integer.parseInt(tfId.getText())+"'";

int result = JOptionPane.showOptionDialog(null, "Are you sure you want to delete this record","Delete Record",JOptionPane.YES\_NO\_OPTION, JOptionPane.QUESTION\_MESSAGE, null, null, null);

if (result == JOptionPane.YES\_OPTION){

for (int i=0;i<data.size();i++)

{

if (data.get(i).getId()==Integer.parseInt(tfId.getText()))

{

stm.execute(query);

data.remove(data.get(i));

employees.remove(employees.get(i));

exist=1;

break;

}

}

}//if (result == JOptionPane.YES\_OPTION)

if (exist==1)

{

tfId.clear();

infoDialog(stage,"Employee Was Successfully Deleted!!!");

}

else

{

popErrorMessage(stage,"Employee Doesn't Exist!!! ");

}

}

/\*\*Error Message \*/

public void popErrorMessage(Stage primaryStage,String text2)

{

/\*\*Create a Stage \*/

final Stage dialog=new Stage ();

dialog.initModality(Modality.APPLICATION\_MODAL);

dialog.initOwner(primaryStage);

/\*\*Set text properties \*/

Text text=new Text();

text.setText(text2);

text.setFill(Color.RED);

text.setFont(Font.font("Times new roman",20));

/\*\*Set image properties \*/

ImageView image =new ImageView(new Image("file:warning.png"));

/\*\*Set the button properties \*/

Button dialogOk=new Button("OK");

dialogOk.setPrefWidth(100);

/\*\*Set dialog HBox properties \*/

HBox dialogHbox=new HBox ();

dialogHbox.setSpacing(10);

dialogHbox.getChildren().addAll(image,text);

HBox.setMargin(image,new Insets(0,0,0,10));

/\*\*Set VBox properties \*/

VBox dialogVBox =new VBox();

dialogVBox.setSpacing(10);

dialogVBox.getChildren().addAll(dialogHbox,dialogOk);

VBox.setMargin(dialogOk,new Insets(0,0,0,170));

/\*\* exit the pop up window\*/

dialogOk.setOnAction(e->{dialog.close();});

/\*\*Create a scene \*/

Scene scene=new Scene (dialogVBox,450,100);

dialog.setScene(scene);

dialog.setTitle("Error Message");

dialog.show();

}

/\*\*Create Information Dialog method \*/

public void infoDialog (Stage primaryStage, String text1)

{

/\*\*Create a Stage zip Files\*/

final Stage dialog=new Stage ();

dialog.initModality(Modality.APPLICATION\_MODAL);

dialog.initOwner(primaryStage);

/\*\*Set text properties \*/

Text localText=new Text();

localText.setText(text1);

localText.setFill(Color.BLUE);

localText.setFont(Font.font("Times new roman",20));

/\*\*Set image properties \*/

ImageView image =new ImageView(new Image("file:info.jpg"));

/\*\*Set dialog HBox properties \*/

HBox dialogHbox=new HBox();

dialogHbox.setSpacing(10);

dialogHbox.getChildren().addAll(image,localText);

HBox.setMargin(image,new Insets(0,0,0,10));

/\*\*Set VBox properties \*/

VBox dialogVBox=new VBox();

Button dialogOk=new Button("Ok");

dialogVBox.setSpacing(10);

dialogVBox.getChildren().addAll(dialogHbox,dialogOk);

VBox.setMargin(dialogOk,new Insets(0,0,0,170));

/\*\*Set the button properties \*/

dialogOk.setPrefWidth(100);

/\*\* exit the pop up window\*/

dialogOk.setOnAction(new EventHandler <ActionEvent>(){

public void handle (ActionEvent event)

{

dialog.close();

}

});

/\*\*Create a scene \*/

Scene scene=new Scene (dialogVBox,450,100);

dialog.setScene(scene);

dialog.setTitle("Info Message");

dialog.show();

}

}//DELETE

//EmployeeManager.java

package empmanager;

/\*\*

\* Application name: Employee manager.

\* Author: Yassine Ballaj.

\* Started Date: 3/23/2015.

\* finished Date:

\*/

import javafx.application.Application;

import javafx.collections.FXCollections;

import javafx.collections.ObservableList;

import javafx.event.Event;

import javafx.event.ActionEvent;

import javafx.event.EventHandler;

import java.sql.\*;

import javafx.scene.input.KeyCode;

import javafx.scene.Scene;

import java.util.ArrayList;

import javafx.geometry.Insets;

import javafx.scene.control.Button;

import javafx.scene.image.Image;

import javafx.scene.image.ImageView;

import javafx.scene.layout.HBox;

import javafx.scene.layout.VBox;

import javafx.scene.paint.Color;

import javafx.scene.text.Font;

import javafx.scene.text.Text;

import javafx.stage.Modality;

import javafx.stage.Stage;

/\*\*

\*

\* @author ballaj

\*/

public class EmployeeManager extends Application {

private SearchPane searchPane =new SearchPane();

private Connection conn;

private Statement stm;

private int [] startingPoint=new int []{0};

private ArrayList<Employee> employees =new ArrayList<Employee>();

@Override

public void start(Stage primaryStage) {

/\*\*Connect to a database \*/

connectToDataBase();

listAllEmployee();

employees.addAll(searchPane.data);

/\*\*List all Employee in the database \*/

searchPane.listAllButton.setOnAction(new EventHandler<ActionEvent>() {

public void handle(ActionEvent event) {

searchPane.data.clear();

searchPane.data.addAll(employees);

}

});

/\*\*Add Employee \*/

searchPane.addEmployee.setOnAction(new EventHandler<ActionEvent>(){

public void handle (ActionEvent event)

{

new AddEmployee(primaryStage,stm, searchPane.data, startingPoint,employees);

}

});

/\*\*Delete employee \*/

searchPane.deleteEmployee.setOnAction(e->{

new deleteEmployee(primaryStage,stm,searchPane.data,employees);

});

/\*\*Update Employee \*/

searchPane.updateEmployee.setOnAction(e->{new UpdateEmployee(primaryStage, stm, searchPane.data, startingPoint,employees);});

/\*\*Search \*/

searchPane.setOnKeyPressed(e->{

if(e.getCode()==KeyCode.ENTER)

{

search(primaryStage);

}

});

/\*\*Exit the application\*/

searchPane.exit.setOnAction(new EventHandler<ActionEvent>(){

public void handle(ActionEvent event)

{

System.exit(0);

}

});

/\*\*Create a scene \*/

Scene scene =new Scene(searchPane,610,500);

primaryStage.setScene(scene);

primaryStage.setTitle("Employee Manager");

primaryStage.show();

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String[] args) {

launch(args);

}

public void connectToDataBase()

{

try {

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

conn=DriverManager.getConnection("jdbc:mysql://localhost:3306/EmployeeManager","root","");

stm=conn.createStatement();

}catch(Exception exception)

{

System.out.println(exception.getMessage());

}

}

public void listAllEmployee()

{

try{

String query ="select \* from employee where id>'"+startingPoint[0]+"'";

ResultSet resultSet=stm.executeQuery(query);

while(resultSet.next())

{

searchPane.data.add(new Employee(

Integer.parseInt(resultSet.getString("id")),

resultSet.getString("firstname"),

resultSet.getString("lastname"),

resultSet.getString("middlename"),

resultSet.getString("position"),

resultSet.getDouble("salary")

));

startingPoint[0]=Integer.parseInt(resultSet.getString(1));

}

}catch(Exception exception)

{

System.out.println(exception.getMessage());

}

}

/\*\*Search method \*/

public void search(Stage primaryStage)

{

int index=-1;

int exist=0;

ArrayList<Integer> indexes=new ArrayList<>();

if (searchPane.searchField.getText().equals(""))

{

searchPane.searchField.requestFocus();

}

else

{

if (searchPane.comboBoxSearch.getValue().equals("Id"))

{

for(int i=0; i<employees.size();i++)

{

if (employees.get(i).getId()==Integer.parseInt(searchPane.searchField.getText()))

{

exist=1;

index=i;

}

}

if (exist==1)

{

searchPane.data.clear();

searchPane.data.add(employees.get(index));

startingPoint[0]=0;

}

else

{

popErrorMessage(primaryStage,"Employee Does Not Exist !!!");

}

}

else if (searchPane.comboBoxSearch.getValue().equals("First name"))

{

for(int i=0; i<employees.size();i++)

{

if (employees.get(i).getFirstName().equals(searchPane.searchField.getText()))

{

exist=1;

indexes.add(i);

}

}

if (exist==1)

{

searchPane.data.clear();

for(int i=0; i<indexes.size();i++)

{

searchPane.data.add(employees.get(indexes.get(i)));

}

startingPoint[0]=0;

}

else

{

popErrorMessage(primaryStage,"Employee Does Not Exist!!!");

}

}

else if (searchPane.comboBoxSearch.getValue().equals("Middle name"))

{

for(int i=0; i<employees.size();i++)

{

if (employees.get(i).getMiddleName().equals(searchPane.searchField.getText()))

{

exist=1;

indexes.add(i);

}

}

if (exist==1)

{

searchPane.data.clear();

for(int i=0; i<indexes.size();i++)

{

searchPane.data.add(employees.get(indexes.get(i)));

}

startingPoint[0]=0;

}

else

{

popErrorMessage(primaryStage,"Employee Does Not Exist!!!");

}

}

else if (searchPane.comboBoxSearch.getValue().equals("Last name"))

{

for(int i=0; i<employees.size();i++)

{

if (employees.get(i).getLastName().equals(searchPane.searchField.getText()))

{

exist=1;

indexes.add(i);

}

}

if (exist==1)

{

searchPane.data.clear();

for(int i=0; i<indexes.size();i++)

{

searchPane.data.add(employees.get(indexes.get(i)));

}

startingPoint[0]=0;

}

else

{

popErrorMessage(primaryStage,"Employee Does Not Exist!!!");

}

}

else if (searchPane.comboBoxSearch.getValue().equals("Position"))

{

for(int i=0; i<employees.size();i++)

{

if (employees.get(i).getPosition().equals(searchPane.searchField.getText()))

{

exist=1;

indexes.add(i);

}

}

if (exist==1)

{

searchPane.data.clear();

for(int i=0; i<indexes.size();i++)

{

searchPane.data.add(employees.get(indexes.get(i)));

}

startingPoint[0]=0;

}

else

{

popErrorMessage(primaryStage,"Employee Does Not Exist!!!");

}

}

else if (searchPane.comboBoxSearch.getValue().equals("Salary"))

{

for(int i=0; i<employees.size();i++)

{

if (employees.get(i).getSalary()==Double.parseDouble(searchPane.searchField.getText()))

{

exist=1;

indexes.add(i);

}

}

if (exist==1)

{

searchPane.data.clear();

for(int i=0; i<indexes.size();i++)

{

searchPane.data.add(employees.get(indexes.get(i)));

}

startingPoint[0]=0;

}

else

{

popErrorMessage(primaryStage,"Employee Does Not Exist!!!");

}

}

}

}

/\*\*Error Message \*/

public void popErrorMessage(Stage primaryStage,String text2)

{

/\*\*Create a Stage \*/

final Stage dialog=new Stage ();

dialog.initModality(Modality.APPLICATION\_MODAL);

dialog.initOwner(primaryStage);

/\*\*Set text properties \*/

Text text=new Text();

text.setText(text2);

text.setFill(Color.RED);

text.setFont(Font.font("Times new roman",20));

/\*\*Set image properties \*/

ImageView image =new ImageView(new Image("file:warning.png"));

/\*\*Set the button properties \*/

Button dialogOk=new Button("OK");

dialogOk.setPrefWidth(100);

/\*\*Set dialog HBox properties \*/

HBox dialogHbox=new HBox ();

dialogHbox.setSpacing(10);

dialogHbox.getChildren().addAll(image,text);

HBox.setMargin(image,new Insets(0,0,0,10));

/\*\*Set VBox properties \*/

VBox dialogVBox =new VBox();

dialogVBox.setSpacing(10);

dialogVBox.getChildren().addAll(dialogHbox,dialogOk);

VBox.setMargin(dialogOk,new Insets(0,0,0,170));

/\*\* exit the pop up window\*/

dialogOk.setOnAction(e->{dialog.close();});

/\*\*Create a scene \*/

Scene scene=new Scene (dialogVBox,450,100);

dialog.setScene(scene);

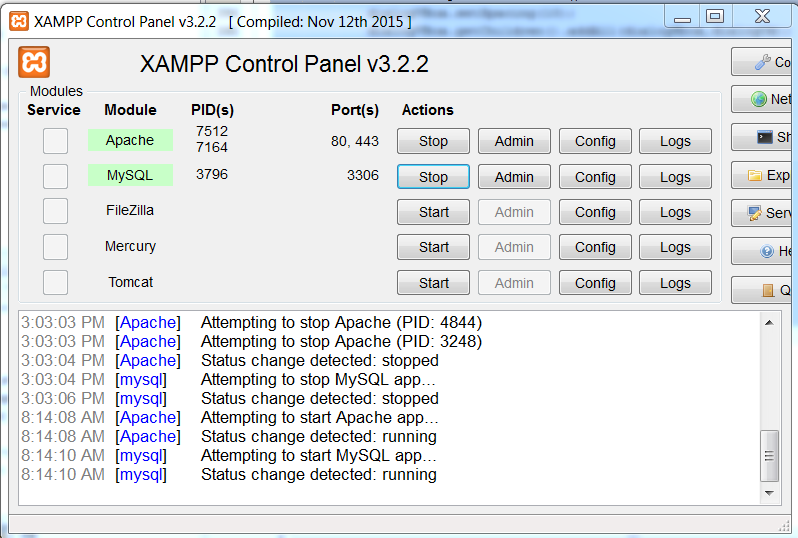
dialog.setTitle("Error Message");

dialog.show();

}

}

Start XAMPP first!!!!



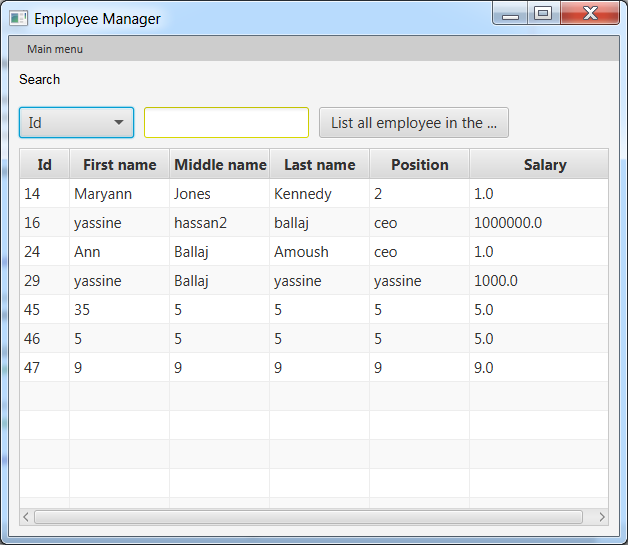
Now in NetBeans ….

F11 to build the project

Then

F6 to run the project

\*\*\*\*The data in your employeemanager.sql may differ from below!!!!



C’est Finis!!!!!!