



Université Abdelmalek Essaâdi  
Faculté des Sciences & Techniques de Tanger  
Département Génie Informatique



# RAPPORT DU PROJET

*Application desktop java*

## Élaboré Par :

EL HAYANI Adnan

Groupe 1

LIEN GITHUB :

[https://github.com/adnanelhayani/project\\_javaFx.git](https://github.com/adnanelhayani/project_javaFx.git)

## Encadré par :

Pr. EL AACHAK LOTFI

Pr. EN-NAIMI EL MOKHTAR

## Remerciement :

Avant Avant tout développement, je tiens à exprimer mes remerciements à mon Pr : EL AACHAK et au Pr : El Mokhtar En-naimi qui m'ont donné l'opportunité de réaliser un travail pratique pour enrichir et développer mes connaissances expérimentales. Je tiens à vous assurer que j'étais très motivé durant la réalisation de ce projet afin de répondre aux attentes de votre part. J'espère que ce travail sera à la hauteur de vos attentes.

## **PLAN DE TRAVAIL:**

- INTRODUCTION GENERAL
- DEFINITION DES ELEMENTS DU TRAVAIL
- Page main
- La connexion à la base de donne
- Le login de page
- Le Menu
- Le premier page
- Deuxième & troisième & quatrième page
- CONCLUSION
- BIBLIOGRAPHIE

### ***Introduction générale***

JavaFX est une bibliothèque logicielle et un framework pour le développement d'interfaces graphiques d'applications de bureau, web et mobiles en utilisant le langage de programmation Java. JavaFX fournit des fonctionnalités pour la création d'interfaces utilisateur graphiques riches et interactives avec des effets visuels avancés, la manipulation d'images et de médias, la prise en charge de CSS pour la conception d'interfaces graphiques et la gestion des événements utilisateur. JavaFX est inclus dans la distribution standard de Java à partir de la version 8 et est multiplateforme, ce qui signifie qu'il fonctionne sur différents systèmes d'exploitation tels que Windows, MacOS et Linux.

## But de ce rapport :

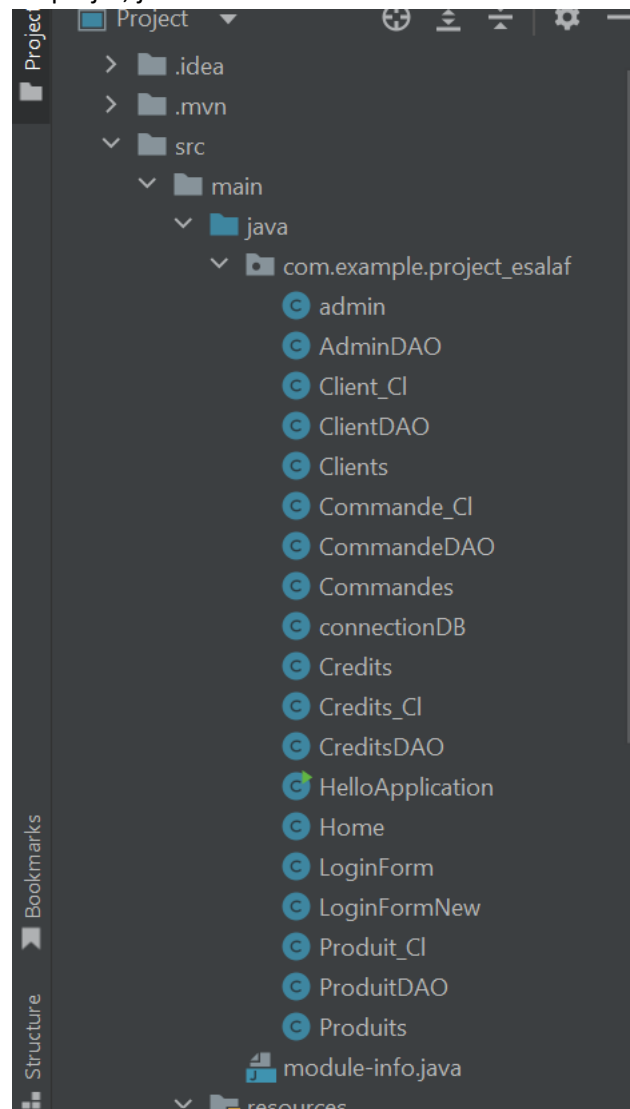
Je montre comment je créer une application desktop java, basée sur JDBC et JavaFX, l'application doit gérer les crédit, les commandes, les produits, les clients, ainsi elle donne un tableau de board générique.

Outils : JavaFX, JDBC, Mysql , Java.

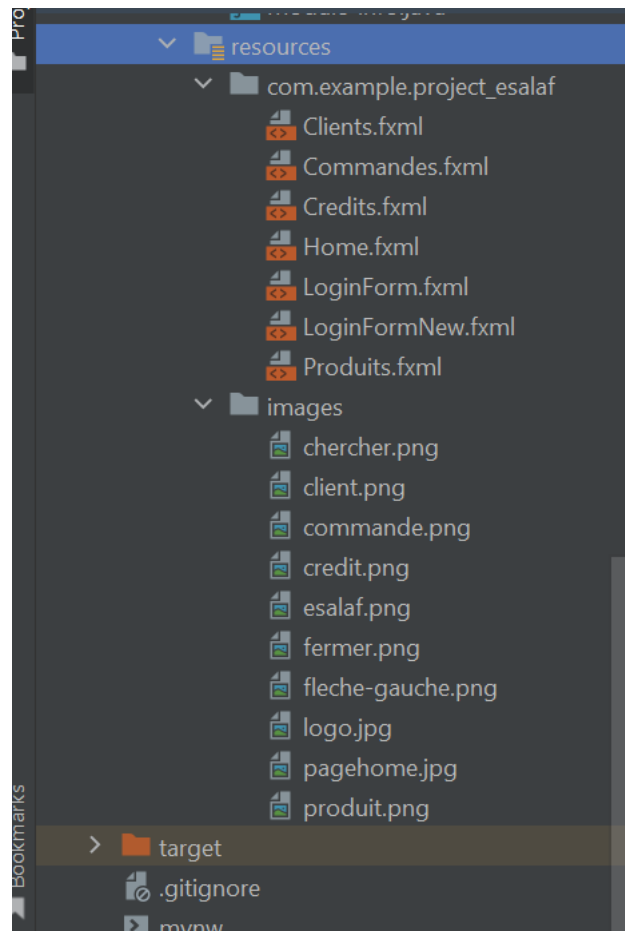
## NOTE :

Avant de passer au code, il est souhaitable de prendre un peu de recul et accorder quelques instants à la compréhension de l'organisation de mes classes.

Après la création de mon projet, je crée 19 classes.



Et 6 fichiers de type fxml



=> **connexionDb** : le classe pour le but de connexion a base de donne.

=>**admin.java & AdminDAO.java & LoginForm.java& LoginForm.fxml &LoginFormNew.java & LoginFormNew.fxml**: Où je créé mon interface de connexion pour accède au menu de l'application.

=>**Home.java&Home.fxml** : Où je créer le menu de l'application.

=>**Produits.java & ProduitDAO.java & Produit\_Cl.java& Produits.fxml** : Où je développé le première tableau(**produits**) de mon application

=>**Clients.java & ClientDAO.java & Client\_Cl.java& Clients.fxml** : Où je développé le deuxième tableau(**clients**) de mon application

=>**Commandes.java & CommandeDAO.java & Commande\_Cl.java& Commandes.fxml** : Où je développé le troisième tableau(**commandes**) de mon application

=>**Credits.java & Credit DAO.java & Credit\_Cl.java& Credits.fxml** : Où je développé le deuxième tableau(**credits**) de mon application

*Classe main de mon application :*

## *HelloApplication.java*

C'est la page d'exécution d'application a paritaire la fonction start et main.

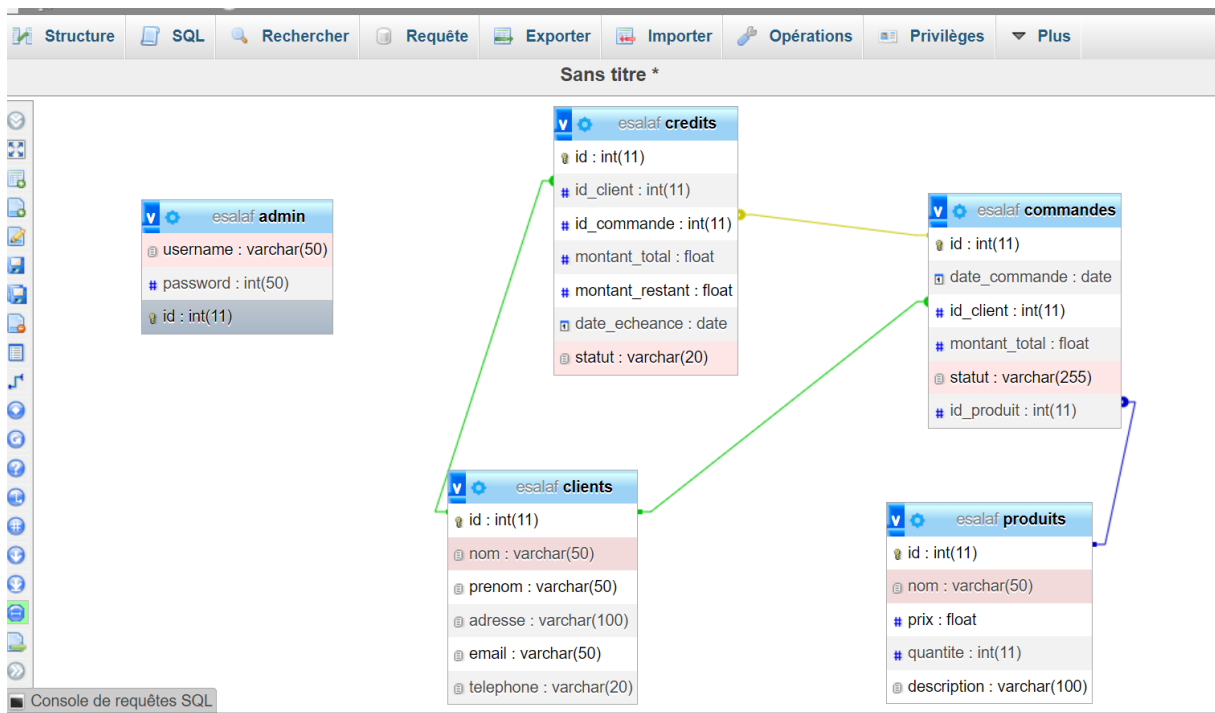
```
package com.example.project_esalaf;
import javafx.application.Application;
import javafx.fxml.FXMLLoader;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.stage.Stage;
import java.io.IOException;
public class HelloApplication extends Application {
    @Override
    public void start(Stage stage) throws IOException {
        Parent root
=FXMLLoader.load(getClass().getResource("/com/example/project_esalaf/LoginForm.fxml"))
;
        stage.setTitle("project application");
        Scene scene = new Scene(root);
        stage.setScene(scene);
        stage.show();
    }
    public static void main(String[] args) {
        launch();
    }
}
```

## Premièrement : La connexion

**connexionDB.java** dans cette classe je créer deux fonctions la première pour ouvrir la connexion à la base de donne pour ça j'utilise la fonction getConnection avec DriverManager qui gérer cette connexion et la deuxième pour fermer cette connexion

```
package com.example.project_esalaf;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.SQLException;
public class connexionDB {
    private static Connection con=null;
    private connexionDB() {
    }
    public static Connection openConnection() throws SQLException {
        if(con==null)
            con= (Connection)
DriverManager.getConnection("jdbc:mysql://127.0.0.1:3306/esalaf","root","");
        return con;
    }
    public static void closeConnection() {
        if(con!=null)
            con=null;
    }
}
```

## *Voila le schème de base de donne.*



## Deuxièmement : Login application

### **A : LoginForm.fxml**

Dans cette classe je crée le page de admin avec un design simple par scene builder et un formulaire qui est contient le prénom et le password pour la connexion.

```
<?xml version="1.0" encoding="UTF-8"?>

<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.control.PasswordField?>
<?import javafx.scene.control.TextField?>
<?import javafx.scene.image.Image?>
<?import javafx.scene.image.ImageView?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.layout.Pane?>
<?import javafx.scene.text.Font?>

<AnchorPane prefHeight="551.0" prefWidth="857.0"
  xmlns="http://javafx.com/javafx/19"
  xmlns:fx="http://javafx.com/fxml/1"
  fx:controller="com.example.project_esalaf.LoginForm">
  <children>
    <Pane prefHeight="482.0" prefWidth="297.0" style="-fx-
```

```

background-color: orange;" AnchorPane.bottomAnchor="0.0"
AnchorPane.leftAnchor="0.0" AnchorPane.topAnchor="0.0">
    <children>
        <ImageView fitHeight="189.0" fitWidth="200.0"
layoutX="49.0" layoutY="64.0" pickOnBounds="true"
preserveRatio="true">
            <image>
                <Image url="@../../images/logo.jpg" />
            </image>
        </ImageView>
        <Label layoutX="52.0" layoutY="297.0" text="Mon
Supermarket">
            <font>
                <Font name="System Italic" size="25.0" />
            </font>
        </Label>
    </children>
</Pane>

<Pane layoutX="400.0" layoutY="68.0" prefHeight="314.0"
prefWidth="330.0" style="-fx-border-style: solid; -fx-border-
color: red;">
    <children>
        <Label layoutX="25.0" layoutY="4.0" text="Le prenom"
textFill="#568f67">
            <font>
                <Font name="System Bold" size="13.0" />
            </font>
        </Label>
        <TextField fx:id="textUser" layoutX="22.0"
layoutY="29.0" prefHeight="36.0" prefWidth="242.0"
promptText="Entrer le nom" />
        <PasswordField fx:id="testPass" layoutX="25.0"
layoutY="115.0" prefHeight="36.0" prefWidth="242.0"
promptText="Entrer le mot de passe" />
        <Button fx:id="btnSign" layoutX="42.0" layoutY="203.0"
mnemonicParsing="false" onAction="#isSign"
prefHeight="25.0" prefWidth="200.0" style="-fx-background-
color: red;" text="Se connecter" />
        <Button layoutX="44.0" layoutY="245.0"
mnemonicParsing="false" prefHeight="25.0"
prefWidth="200.0" style="-fx-background-color: orange;"
text="S'inscrire" />
        <Label fx:id="lblErr" layoutX="38.0" layoutY="164.0"
prefHeight="17.0" prefWidth="217.0" textFill="#a12006" />
        <Label layoutX="25.0" layoutY="96.0" text="Le mot de
passe " textFill="#568f67">

```

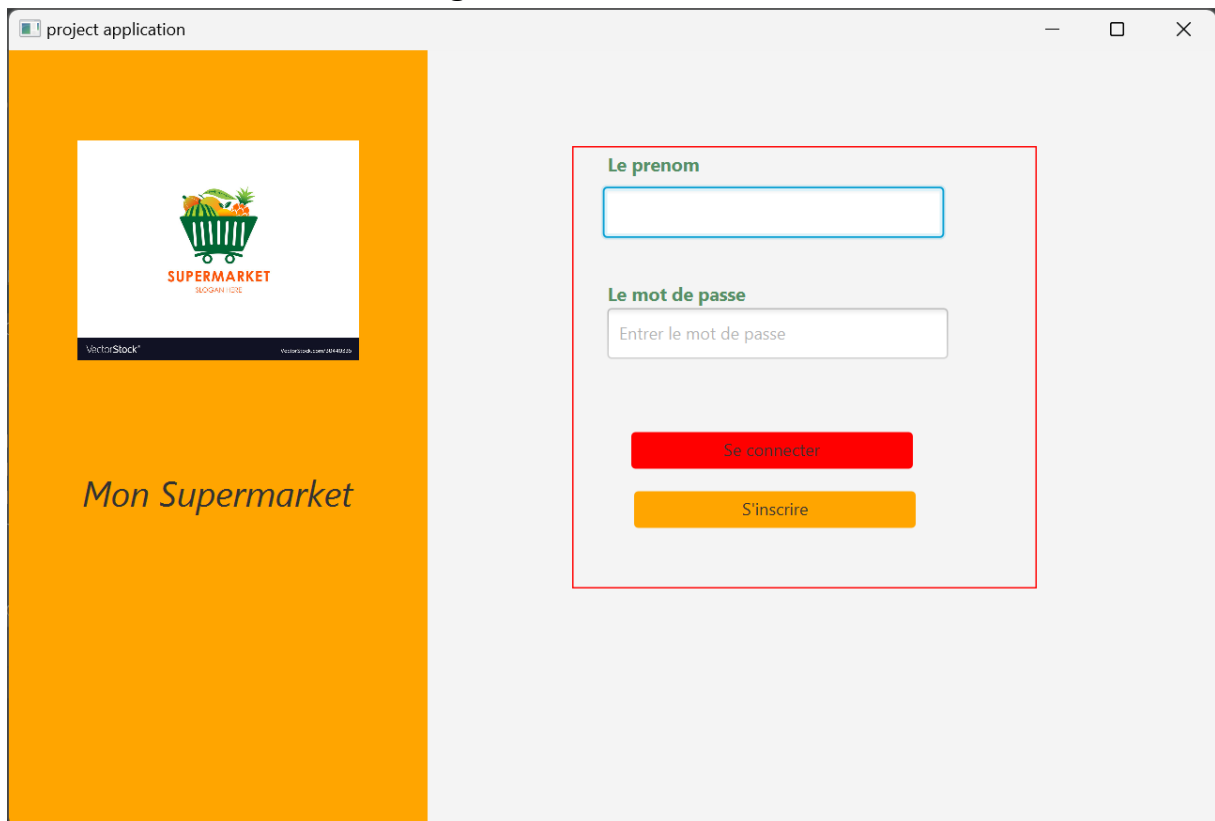


```

<font>
  <Font name="System Bold" size="13.0" />
</font>
</Label>
</children>
</Pane>
</children>
</AnchorPane>

```

Et voila le résultat de ce design :



## ***B : admin.java***

Dans cette classe je déclare deux attributs (username, password) qui sont correspondant au tableau admin qui est existe dans base de donne mysql avec les getters and les setters

```

package com.example.project_esalaf;
public class admin {
    private String username;
    private String password;
    public String getUsername() {
        return username;
    }
    public void setUsername(String username) {
        this.username = username;
    }
    public String getPassword() {
        return password;
    }
    public void setPassword(String password) {
        this.password = password;
    }
}

```

### ***C : AdminDAO.java***

Dans cette classe je teste à partiaire la fonction isLogin les informations qui est saisi dans le formulaire avec les informations qui existe dans le tableau admin dans base de donne si les information et vrai la fonction retourne true sinon retourne false.

```

package com.example.project_esalaf;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
public class AdminDAO {
    Statement stm;
    public boolean isLogin(admin adm) throws SQLException {
        stm=connectionDB.openConnection().createStatement();
        ResultSet res= stm.executeQuery(" select * from admin where
username='"+adm.getUsername()+"' and password='"+adm.getPassword()+"'");
        if (res.next()){
            return true;
        }
        return false;
    }
}

```

### ***D: LoginForm.java***

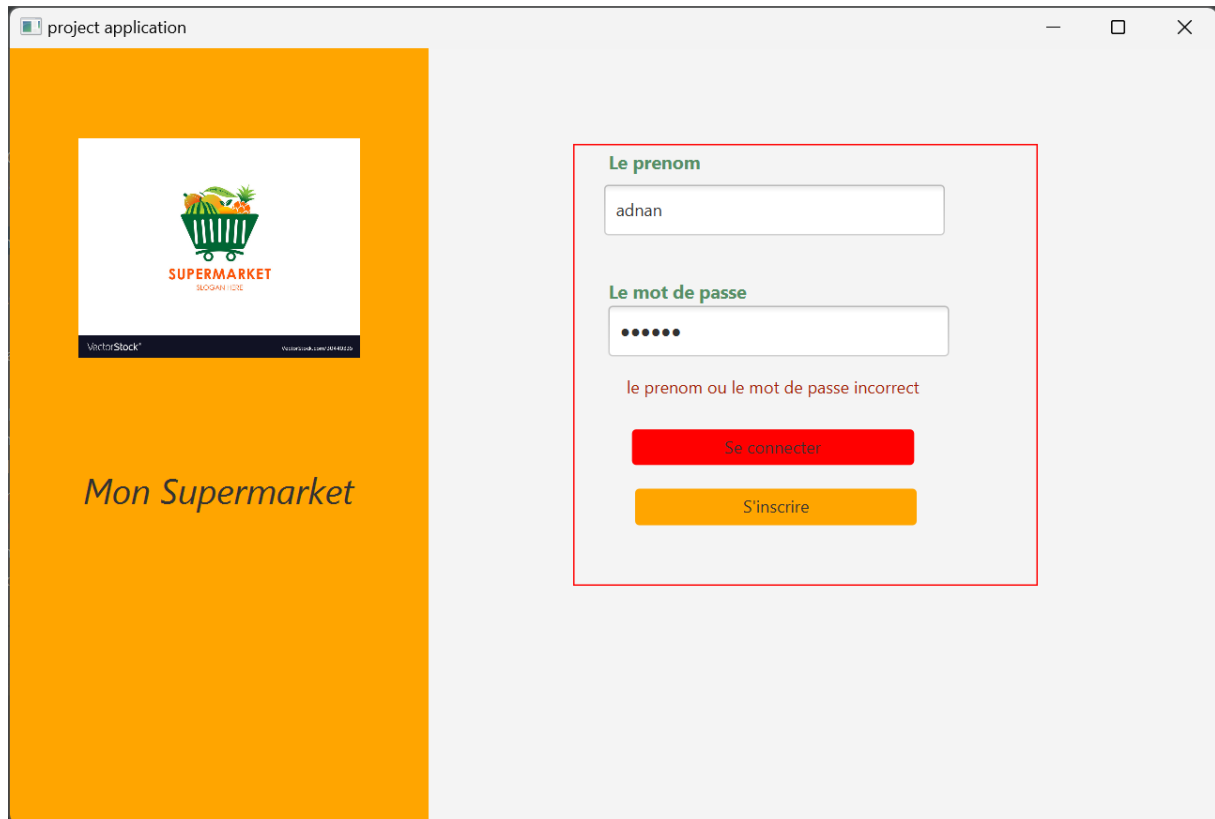
Dans cette classe je ouvre l'interface de page home à partiaire la fonction isSign si la fonction isLogin retourne true et je click sur la Button se connecter la page home est ouvre sinon il afficher un message erreur qui contient se phrase (le prénom ou le mot de passe incorrect).

```

package com.example.project_esalaf;
import javafx.event.ActionEvent;
import javafx.fxml.FXMLLoader;
import javafx.fxml.FXML;
import javafx.scene.Node;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.scene.control.Label;
import javafx.scene.control.PasswordField;
import javafx.scene.control.TextField;
import javafx.stage.Stage;
import java.io.IOException;
import java.sql.SQLException;
public class LoginForm {
    @FXML
    public TextField textUser;
    public PasswordField testPass;
    public Button btnSign;
    public Label lblErr;
    admin adm=new admin();
    AdminDAO lg=new AdminDAO();
    public LoginForm() {
        textUser = new TextField();
        testPass = new PasswordField();
        lblErr = new Label();
    }
    public void isSign(ActionEvent actionEvent) throws SQLException, IOException {
        adm.setUsername(textUser.getText());
        adm.setPassword(testPass.getText());
        if(lg.isLogin(adm)){
            Node node =(Node) actionEvent.getSource();
            Stage stage=(Stage) node.getScene().getWindow();
            Parent root
=FXMLLoader.load(getClass().getResource("/com/example/project_esalaf/Home.fxml"));
            Scene scene = new Scene(root);
            stage.setScene(scene);
            stage.show();
        }else {
            lblErr.setText("le nom ou le password incorrect");
        }
    }
}

```

**Voila le resultat :**



### ***E: FormLoginNew.java***

Dans cette classe je crée deux méthodes, **loginAdmin** qui me permet de retourner au page précédent de login et **handleAddadmin** cette fonction pour le but d'ajouter un nouveau admin a l'application .

```
package com.example.project_esalaf;
import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.scene.Node;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.control.*;
import javafx.stage.Stage;
import java.io.IOException;
import java.sql.SQLException;
import java.sql.Statement;
public class LoginFormNew {
    @FXML
    public TextField textUser;
    public PasswordField testPass;
    public Button btnSign;
```

```

public Label lblErr;
admin adm=new admin();
AdminDAO lg=new AdminDAO();
public LoginFormNew() {
    textUser = new TextField();
    testPass = new PasswordField();
    lblErr = new Label();
}
public boolean addAdmin(admin adm) throws SQLException {
    Statement stm = connectionDB.openConnection().createStatement();
    String query = "INSERT INTO admin (username, password) VALUES ('" + adm.getUsername()
+ "', '" + adm.getPassword() + "')";
    int rowsInserted = stm.executeUpdate(query);
    if (rowsInserted > 0) {
        return true;
    }
    return false;
}
@FXML
public void handleAddAdmin(ActionEvent event) throws SQLException, IOException {
    // Création d'un nouvel objet admin
    admin adm = new admin(textUser.getText(), Integer.parseInt(testPass.getText()));
    boolean result =addAdmin(adm);
    if (result) {
        Alert alert = new Alert(Alert.AlertType.INFORMATION);
        alert.setTitle("Succès");
        alert.setHeaderText(null);
        alert.setContentText("Administrateur ajouté avec succès !");
        alert.showAndWait();
        Node node =(Node) event.getSource();
        Stage stage=(Stage) node.getScene().getWindow();
        Parent root =
FXMLLoader.load(getClass().getResource("/com/example/project_esalaf/LoginForm.fxml"));
        Scene scene = new Scene(root);
        stage.setScene(scene);
        stage.show();
    } else {
        Alert alert = new Alert(Alert.AlertType.ERROR);
        alert.setTitle("Erreur");
        alert.setHeaderText(null);
        alert.setContentText("Erreur lors de l'ajout de l'administrateur !");
        alert.showAndWait();
    }
}

public void loginAdmin(ActionEvent actionEvent) throws IOException {
    Node node =(Node) actionEvent.getSource();
    Stage stage=(Stage) node.getScene().getWindow();

```

```
Parent root =  
FXMLLoader.load(getClass().getResource("/com/example/project_esalaf/LoginForm.fxml"));  
Scene scene = new Scene(root);  
stage.setScene(scene);  
stage.show();  
}  
}
```

**Voila la page :**

project application

## Nouveau admin

Le prenom

Le mot de passe

Entrer le mot de passe

Ajouter

Annuler

## Troisièmement : page home.

### **A : Home.fxml**

Dans cette classe je crée la page de home avec un design simple par scene builder et un menu qui est quatre élément (produits, clients, commandes, crédits).

```
<?xml version="1.0" encoding="UTF-8"?>

<?import javafx.geometry.Insets?>
<?import javafx.scene.Cursor?>
<?import javafx.scene.control.Button?>
<?import javafx.scene.control.Label?>
<?import javafx.scene.image.Image?>
<?import javafx.scene.image.ImageView?>
<?import javafx.scene.layout.AnchorPane?>
<?import javafx.scene.layout.Pane?>
<?import javafx.scene.layout.VBox?>
<?import javafx.scene.text.Font?>

<AnchorPane prefHeight="551.0" prefWidth="860.0" style="-fx-background-color: orange;"
xmlns="http://javafx.com/javafx/19" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="com.example.project_esalaf.Home">
    <children>
        <Pane cacheShape="false" centerShape="false" layoutX="234.0" prefHeight="400.0"
prefWidth="365.0" scaleShape="false">
            <children>
                <Pane layoutX="6.0" layoutY="-116.0" prefHeight="200.0" prefWidth="131.0"
rotate="-42.6" style="-fx-background-color: red; -fx-background-radius: 50;" />
            </children>
            <opaqueInsets>
                <Insets />
            </opaqueInsets>
        </Pane>
        <Pane layoutY="-12.0" prefHeight="563.0" prefWidth="871.0"
AnchorPane.bottomAnchor="0.0" AnchorPane.leftAnchor="0.0" AnchorPane.rightAnchor="-
11.0" AnchorPane.topAnchor="-12.0">
            <children>
                <Pane layoutX="14.0" layoutY="75.0" prefHeight="42.0" prefWidth="200.0" style="-
fx-background-color: FA2C56; -fx-background-radius: 15;">
                    <children>
                        <Label layoutX="42.0" layoutY="8.0" text="Page Admin" textFill="WHITE">
                            <font>
                                <Font size="18.0" />
                            </font>
                        </Label>
                    </children>
                </Pane>
            </children>
        </Pane>
    </children>
</AnchorPane>
```

```

</Pane>
<VBox layoutX="45.0" layoutY="174.0" prefHeight="375.0" prefWidth="273.0">
  <children>
    <Button fx:id="btnProduit" graphicTextGap="40.0" mnemonicParsing="false"
onAction="#openProduit" prefHeight="59.0" prefWidth="233.0" style="-fx-background-
color: orange;" text="Produits">
      <graphic>
        <ImageView cache="true" fitHeight="50.0" fitWidth="80.0"
nodeOrientation="INHERIT" pickOnBounds="true" preserveRatio="true">
          <image>
            <Image url="@../././images/produit.png" />
          </image>
        </ImageView>
      </graphic>
      <cursor>
        <Cursor fx:constant="HAND" />
      </cursor>
    </Button>
    <Button fx:id="btnClient" graphicTextGap="42.0" mnemonicParsing="false"
onAction="#openClient" prefHeight="60.0" prefWidth="232.0" style="-fx-background-color:
orange;" text="Clients ">
      <graphic>
        <ImageView fitHeight="50.0" fitWidth="59.0" pickOnBounds="true"
preserveRatio="true">
          <image>
            <Image url="@../././images/client.png" />
          </image>
        </ImageView>
      </graphic></Button>
    <Button fx:id="btnCommande" graphicTextGap="37.0" mnemonicParsing="false"
onAction="#openCommande" prefHeight="66.0" prefWidth="237.0" style="-fx-background-
color: orange;" text="Commandes">
      <graphic>
        <ImageView fitHeight="43.0" fitWidth="55.0" pickOnBounds="true"
preserveRatio="true">
          <image>
            <Image url="@../././images/commande.png" />
          </image>
        </ImageView>
      </graphic></Button>
    <Button fx:id="btnCredit" graphicTextGap="40.0" mnemonicParsing="false"
onAction="#openCredit" prefHeight="67.0" prefWidth="240.0" style="-fx-background-color:
orange;" text="Credits ">
      <graphic>
        <ImageView fitHeight="46.0" fitWidth="49.0" pickOnBounds="true"
preserveRatio="true">
          <image>

```

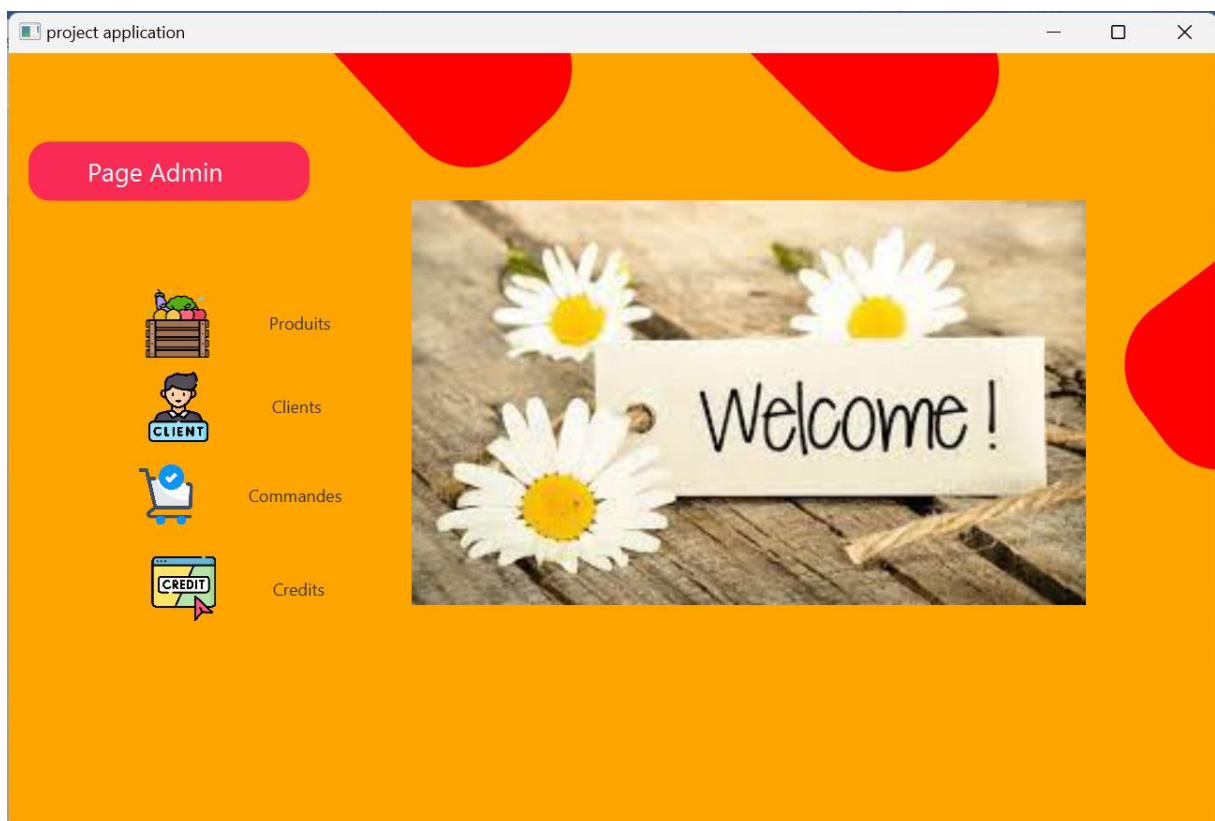


```

        <Image url="@../././images/credit.png" />
    </image>
</ImageView>
</graphic></Button>
</children>
</VBox>
<Pane layoutX="543.0" layoutY="-100.0" prefHeight="200.0" prefWidth="131.0"
rotate="-45.0" style="-fx-background-color: red; -fx-background-radius: 50;" />
    <Pane layoutX="828.0" layoutY="116.0" prefHeight="200.0" prefWidth="131.0"
rotate="53.1" style="-fx-background-color: red; -fx-background-radius: 50;" />
</children>
</Pane>
</children>
</AnchorPane>

```

**Et voila le résultat obtenu :**



## ***B:Home.java***

Dans cette classe je crée quatre fonctions openProduit qui permet d'ouvrir la page produite quand on click sur le bouton produits dans la page home, openClient qui permet d'ouvrir la page clients

quand on click sur le bouton clients dans la page home, openComande qui permet d'ouvrir la page commandes quand on click sur le bouton commandes dans la page home et openCredits qui permet d'ouvrir la page crédits quand on click sur le bouton crédits dans la page home.

```
package com.example.project_esalaf;
import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.scene.Node;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.control.Button;
import javafx.stage.Stage;
import java.io.IOException;
public class Home {
    @FXML
    Button btnProduit, btnClient, btnCommande, btnCredit;
    public void openProduit(ActionEvent actionEvent) throws IOException {
        Node node =(Node) actionEvent.getSource();
        Stage stage=(Stage) node.getScene().getWindow();
//        Parent root =
FXMLLoader.load(getClass().getResource("/com/example/project_esalaf/Home.fxml"));
        Parent root =
FXMLLoader.load(getClass().getResource("/com/example/project_esalaf/Produits.fxml"));
        Scene scene = new Scene(root);
        stage.setScene(scene);
        stage.show();
    }
    public void openClient(ActionEvent actionEvent) throws IOException {
        Node node =(Node) actionEvent.getSource();
        Stage stage=(Stage) node.getScene().getWindow();
//        Parent root =
FXMLLoader.load(getClass().getResource("/com/example/project_esalaf/Home.fxml"));
        Parent root =
FXMLLoader.load(getClass().getResource("/com/example/project_esalaf/Clients.fxml"));
        Scene scene = new Scene(root);
        stage.setScene(scene);
        stage.show();
    }
    public void openCommande(ActionEvent actionEvent) throws IOException {
        Node node =(Node) actionEvent.getSource();
        Stage stage=(Stage) node.getScene().getWindow();
//        Parent root =
FXMLLoader.load(getClass().getResource("/com/example/project_esalaf/Home.fxml"));
        Parent root =
FXMLLoader.load(getClass().getResource("/com/example/project_esalaf/Commandes.fxml"));
    }
}
```

```

        Scene scene = new Scene(root);
        stage.setScene(scene);
        stage.show();
    }
    public void openCredit(ActionEvent actionEvent) throws IOException {
        Node node =(Node) actionEvent.getSource();
        Stage stage=(Stage) node.getScene().getWindow();
        //      Parent root =
        FXMLLoader.load(getClass().getResource("/com/example/project_esalaf/Home.fxml"));
        Parent root =
        FXMLLoader.load(getClass().getResource("/com/example/project_esalaf/Credits.fxml"));
        Scene scene = new Scene(root);
        stage.setScene(scene);
        stage.show();
    }
}

```

## Quatrièmement : pages produits

### **A : Produits. fxml**

Dans cette classe je crée la page de produits avec un design simple par scene builder qui contient un bouton pour routeur au page home et autre bouton pour quitte l'exécution et une barre de recherche avec le bouton de recherche ,un tableau correspondant au tableau produits qui existe dans BDD , un formulaire correspondant au tableau et autres trois boutons pour ajouter, modifier et supprimer les information du tableau .

```

<?xml version="1.0" encoding="UTF-8"?>
<?import javafx.scene.control.*?>
<?import javafx.scene.image.*?>
<?import javafx.scene.layout.*?>
<?import javafx.scene.text.*?>
<AnchorPane prefHeight="552.0" prefWidth="1022.0"
xmlns="http://javafx.com/javafx/17.0.2-ea" xmlns:fx="http://javafx.com/fxml/1"
fx:controller="com.example.project_esalaf.Produit_CI">
    <children>
        <Pane prefHeight="552.0" prefWidth="333.0" style="-fx-background-color: orange;">
            <children>
                <Pane layoutX="72.0" layoutY="21.0" prefHeight="65.0" prefWidth="200.0">
                    <children>
                        <Label fx:id="lb" layoutX="36.0" layoutY="18.0" text="Page Produits "
textFill="WHITE">

```

```

        <font>
            <Font size="20.0" />
        </font>
    </Label>
</children>
</Pane>
<Pane layoutX="2.0" layoutY="79.0" prefHeight="477.0" prefWidth="303.0">
    <children>
        <TextField fx:id="testnom" layoutX="24.0" layoutY="22.0" prefHeight="38.0"
prefWidth="265.0" promptText="entrer le nom de produit" />
        <TextField fx:id="testdescription" layoutX="24.0" layoutY="164.0"
prefHeight="38.0" prefWidth="265.0" promptText="entrer une description sur le produit"
/>
        <TextField fx:id="testprix" layoutX="24.0" layoutY="69.0" prefHeight="38.0"
prefWidth="265.0" promptText="entrer le prix de produit" />
        <TextField fx:id="testquantite" layoutX="23.0" layoutY="115.0" prefHeight="38.0"
prefWidth="265.0" promptText="entrer la quantite de produit" />
        <Button fx:id="btnajouter" layoutX="115.0" layoutY="256.0"
mnemonicParsing="false" onAction="#ajouter" prefHeight="30.0" prefWidth="80.0"
text="Ajouter" />
        <Button fx:id="btnsupprimer" layoutX="115.0" layoutY="299.0"
mnemonicParsing="false" onAction="#supprimer" prefHeight="30.0" prefWidth="80.0"
text="Supprimer" />
        <Button fx:id="btnmodifier" layoutX="115.0" layoutY="347.0"
mnemonicParsing="false" onAction="#modifier" prefHeight="30.0" prefWidth="80.0"
text="Modifier" />
    </children>
</Pane>
</children>
</Pane>
<Pane layoutX="309.0" prefHeight="552.0" prefWidth="715.0">
    <children>
        <Pane layoutX="119.0" layoutY="16.0" prefHeight="65.0" prefWidth="412.0">
            <children>
                <TextField fx:id="testsearch" layoutX="10.0" layoutY="20.0" prefHeight="38.0"
prefWidth="352.0" promptText="entrer le nom de produit " />
            </children>
        </Pane>
        <TableView fx:id="table" layoutX="22.0" layoutY="110.0"
onMouseClicked="#ClickTable" prefHeight="444.0" prefWidth="694.0">
            <columns>
                <TableColumn fx:id="colid" prefWidth="73.33331298828125" text="id" />
                <TableColumn fx:id="colnom" prefWidth="156.6666259765625" text="nom" />
                <TableColumn fx:id="colprix" prefWidth="144.0" text="prix" />
                <TableColumn fx:id="colquantite" prefWidth="146.0" text="quantite" />
                <TableColumn fx:id="coldescription" prefWidth="173.33343505859375"
text="description" />
            </columns>
        </TableView>
    </children>
</Pane>

```

```

        </columns>
    </TableView>
    <Button fx:id="btnreteur" layoutX="27.0" layoutY="21.0" mnemonicParsing="false"
onAction="#retourPagePrecedente" prefHeight="38.0" prefWidth="63.0">
        <graphic>
            <ImageView fitHeight="47.0" fitWidth="98.0" pickOnBounds="true"
preserveRatio="true">
                <image>
                    <Image url="@../../images/fleche-gauche.png" />
                </image>
            </ImageView>
        </graphic></Button>
    <Button fx:id="btnsear" layoutX="560.0" layoutY="14.0" mnemonicParsing="false"
onAction="#search" prefHeight="55.0" prefWidth="53.0" style="-fx-background-radius:
22;">
        <graphic>
            <ImageView fitHeight="54.0" fitWidth="55.0" pickOnBounds="true"
preserveRatio="true">
                <image>
                    <Image url="@../../images/chercher.png" />
                </image>
            </ImageView>
        </graphic></Button>
    </children>
</Pane>
    <Button layoutX="24.0" layoutY="24.0" mnemonicParsing="false"
onAction="#quitterApplication" prefHeight="38.0" prefWidth="34.0">
        <graphic>
            <ImageView fitHeight="38.0" fitWidth="49.0" pickOnBounds="true"
preserveRatio="true">
                <image>
                    <Image url="@../../images/fermer.png" />
                </image>
            </ImageView>
        </graphic>
    </Button>

</children>
</AnchorPane>

```

**Voilà le résultat :**

id	nom	prix	quantite	description
1	farine	10.0	5	Pain et céréales(farine)
2	pain	5.0	4	Pain et céréales(pain)
3	sucre,pain	11.0	1	sucre
4	poisson,thé	10.0	2	Viande et poisson(poisson)
5	thé	15.0	1	thé
6	thon	18.0	2	Produits en conserve(thon)

## **B : Produits. Java**

Dans cette classe je déclare tous les attributs (id,quantite,prix,nom,description) qui sont correspondant au tableau produits qui est existe dans base de donne mysql avec les getters and les setters and les constructeurs

```
package com.example.project_esalaf;

public class Produits {
    private int id ,quantite;
    private Float prix;
    private String nom,description;
    public Produits() {
        this.id = 0;
        this.quantite = 0;
        this.prix = null;
        this.nom = "";
        this.description = "";
    }
    public Produits(int id, int quantite, Float prix, String nom, String description) {
        this.id = id;
        this.quantite = quantite;
        this.prix = prix;
        this.nom = nom;
        this.description = description;
    }
    public int getId() {
        return id;
    }
}
```

```

    }
    public void setId(int id) {
        this.id = id;
    }
    public int getQuantite() {
        return quantite;
    }
    public void setQuantite(int quantite) {
        this.quantite = quantite;
    }
    public Float getPrix() {
        return prix;
    }
    public void setPrix(Float prix) {
        this.prix = prix;
    }
    public String getNom() {
        return nom;
    }
    public void setNom(String nom) {
        this.nom = nom;
    }
    public String getDescription() {
        return description;
    }
    public void setDescription(String description) {
        this.description = description;
    }
}

```

## ***B : ProduitDAO. Java***

Dans cette classe je déclare toutes les méthodes qui sont en relation avec BDD et que je veux utiliser dans la classe produit\_CI, donc je déclare et défini la fonction **insert** pour le but d'ajouter les informations au BDD, **update** pour le but de modification de donne, **delete** pour supprimer les donnees par par sont id, **getAllProduits** pour afficher toutes les donnees de tableau produits, **getsearchProduits** pour le but de recherche sur un produit par son nom.

```

package com.example.project_esalaf;
import javafx.collections.FXCollections;
import javafx.collections.ObservableList;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;

```

```

import java.util.logging.Level;
import java.util.logging.Logger;
public class ProduitDAO {
    Statement state;
    public void insert(Produits produit){
        try {
            state= connectionDB.openConnection().createStatement();
            state.executeUpdate("INSERT INTO
`produits`(`nom`,`prix`,`quantite`,`description`)VALUES('"+produit.getNom()+"','"+produit.get
Prix()+"','"+produit.getQuantite()+"','"+produit.getDescription()+"");
            connectionDB.closeConnection();
        } catch (SQLException e) {
            connectionDB.closeConnection();
            Logger.getLogger(ProduitDAO.class.getName()).log(Level.SEVERE,null,e);
        }
    }
    public void delete(int id){
        try {
            state= connectionDB.openConnection().createStatement();
            state.executeUpdate("Delete FROM `produits`WHERE id="+id);
            connectionDB.closeConnection();
        } catch (SQLException e) {
            connectionDB.closeConnection();
            Logger.getLogger(ProduitDAO.class.getName()).log(Level.SEVERE,null,e);
        }
    }
    public void update(Produits produit){
        try {
            state= connectionDB.openConnection().createStatement();
            state.executeUpdate("UPDATE produits SET `nom` = '"+produit.getNom()+"', `prix` =
"+produit.getPrix()+"', `quantite` = '"+produit.getQuantite()+"', `description` =
 '"+produit.getDescription()+"' WHERE id = '"+produit.getId());
            connectionDB.closeConnection();
        } catch (SQLException e) {
            connectionDB.closeConnection();
            Logger.getLogger(ProduitDAO.class.getName()).log(Level.SEVERE,null,e);
        }
    }
    public ObservableList<Produits> getAllProduits() throws SQLException {
        ObservableList pro= FXCollections.observableArrayList();
        try {
            state=connectionDB.openConnection().createStatement();
            ResultSet resul= state.executeQuery("SELECT *FROM produits");
            while (resul.next()){
                Produits obj =new Produits();
                obj.setId(resul.getInt(1));
                obj.setNom(resul.getString(2));
            }
        }
    }
}

```



```

        obj.setPrix(resul.getFloat(3));
        obj.setQuantite(resul.getInt(4));
        obj.setDescription(resul.getString(5));
        pro.add(obj);
    }
    connectionDB.closeConnection();
} catch (SQLException e) {
    Logger.getLogger(ProduitDAO.class.getName()).log(Level.SEVERE,null,e);
}
return pro;
}
public ObservableList<Produits> getSearchProduits(String nom) throws SQLException {

    ObservableList pro= FXCollections.observableArrayList();
    try {
        state=connectionDB.openConnection().createStatement();
        ResultSet resul= state.executeQuery("SELECT *FROM produits WHERE nom LIKE
        '%" +nom+"%'");
        while (resul.next()){
            Produits obj =new Produits();
            obj.setId(resul.getInt(1));
            obj.setNom(resul.getString(2));
            obj.setPrix(resul.getFloat(3));
            obj.setQuantite(resul.getInt(4));
            obj.setDescription(resul.getString(5));
            pro.add(obj);
        }
        connectionDB.closeConnection();
    } catch (SQLException e) {
        Logger.getLogger(ProduitDAO.class.getName()).log(Level.SEVERE,null,e);
    }
    return pro;
}
}

```

## ***C : Produit\_Cl. Java***

Dans cette classe je crée une fonction **search** qui permet de recherche les produits par son nom dans le tableau ce fonction appelle la fonction getsearchProduits, **Ajouter** qui permet de appelle la fonction insert pour ajouter qui existe dans le formulaire au tableau et à BDD au même temp, **modifier** qui permet d'appelle la fonction update pour modifier les informations du tableau, **delete** qui appelle la fonction delete pour supprimer un élément sélection par son id ,**initialize** qui permet d'appelle la fonction getAllProduits pour initialise le tableau de page par les donnees de ce tableau produits qui existe dans BDD, **quitterApplication** le but de cette fonction c'est l'arrête le exécution de programme et

finalement **retourPagePrecedente** ce fonction qui me permet de retourne a page précèdent quand on click sur le buton.

```
package com.example.project_esalaf;
import javafx.event.ActionEvent;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.fxml.Initializable;
import javafx.scene.Node;
import javafx.scene.Parent;
import javafx.scene.Scene;
import javafx.scene.control.*;
import javafx.scene.control.cell.PropertyValueFactory;
import javafx.scene.input.MouseEvent;
import javafx.stage.Stage;
import java.io.IOException;
import java.net.URL;
import java.sql.SQLException;
import java.util.ResourceBundle;
public class Produit_CI implements Initializable {
    @FXML
    private Label lb;
    @FXML
    private TextField testnom;
    @FXML
    private TextField testdescription;
    @FXML
    private TextField testprix;
    @FXML
    private TextField testquantite;
    @FXML
    private Button btnajouter;
    @FXML
    private Button btnsupprimer;
    @FXML
    private Button btnmodifier;
    @FXML
    private Button btnsear;
    @FXML
    private TextField testsearch;
    @FXML
    private TableView table;
    @FXML
    private TableColumn colid;
    @FXML
    private TableColumn colnom;
```

```

@FXML
private TableColumn colprix;
@FXML
private TableColumn colquantite;
@FXML
private TableColumn coldescription;
ProduitDAO pd=new ProduitDAO();
int ID ;
public void search() throws SQLException {
    table.setItems(pd.getSearchProduits(testsearch.getText()));
}
public void ajouter (ActionEvent actionEvent) throws SQLException {
    Produits pp=new Produits();
    pp.setNom( testnom.getText());
    pp.setPrix( Float.parseFloat(testprix.getText()));
    pp.setQuantite( Integer.parseInt(testquantite.getText()));
    pp.setDescription( testdescription.getText());
    pd.insert(pp);
    table.setItems(pd.getAllProduits());
    testnom.setText("");
    testprix.setText("");
    testquantite.setText("");
    testdescription.setText("");
}
public void modifier(ActionEvent actionEvent) throws SQLException {
    Produits pp = new Produits();
    pp.setId(ID); // on récupère l'ID du produit à modifier
    pp.setNom(testnom.getText());
    pp.setPrix(Float.parseFloat(testprix.getText()));
    pp.setQuantite(Integer.parseInt(testquantite.getText()));
    pp.setDescription(testdescription.getText());
    // appel de la méthode update de ProduitDAO pour mettre à jour les informations du
produit dans la base de données
    pd.update(pp);
    // on rafraîchit le tableau avec les produits mis à jour
    table.setItems(pd.getAllProduits());
    // on réinitialise les champs de saisie et on désactive les boutons Modifier et Supprimer
    testnom.setText("");
    testprix.setText("");
    testquantite.setText("");
    testdescription.setText("");
}
public void supprimer(ActionEvent actionEvent) throws SQLException {
    pd.delete(ID);
    table.setItems(pd.getAllProduits());
    testnom.setText("");
    testprix.setText("");
}

```

```

        testquantite.setText("");
        testdescription.setText("");
    }
    @FXML
    public void ClickTable(MouseEvent mouseEvent) {
        // Obtenir le produit sélectionné dans le tableau
        Produits produits = (Produits) table.getSelectionModel().getSelectedItem();
        // Vérifier si un produit a été sélectionné
        if (produits != null) {
            // Afficher les informations du produit sélectionné dans les champs correspondants
            testnom.setText(produits.getNom());
            testprix.setText(produits.getPrix() + "");
            testquantite.setText(produits.getQuantite() + "");
            testdescription.setText(produits.getDescription());
            // Conserver l'ID du produit sélectionné pour la modification ultérieure
            ID = produits.getId();
        }
    }
    public void retourPagePrecedente(ActionEvent event) throws IOException {
        Parent root =
FXMLLoader.load(getClass().getResource("/com/example/project_esalaf/Home.fxml"));
        Scene scene = new Scene(root);
        Stage stage = (Stage) ((Node) event.getSource()).getScene().getWindow();
        stage.setScene(scene);
        stage.show();
    }
    public void quitterApplication(ActionEvent event) {
        Stage stage = (Stage) ((Node) event.getSource()).getScene().getWindow();
        stage.close();
    }
    @Override
    public void initialize(URL location, ResourceBundle resources) {
        colid.setCellValueFactory(new PropertyValueFactory<>("id"));
        colnom.setCellValueFactory(new PropertyValueFactory<>("nom"));
        colprix.setCellValueFactory(new PropertyValueFactory<>("prix"));
        colquantite.setCellValueFactory(new PropertyValueFactory<>("quantite"));
        coldescription.setCellValueFactory(new PropertyValueFactory<>("description"));
        try {
            table.setItems(pd.getAllProduits());
        } catch (SQLException e) {
            throw new RuntimeException(e);
        }
    }
}

```

[illegible]

***Voilà le résultat du tableau 3 : Commandes :***

[illegible]

**Voilà le résultat du tableau 4 : Crédits :**

## ***CONCLUSION GENERAL***

Tout au long de la préparation de notre jeu, nous avons essayé de pratiquer les connaissances requises durant notre cours de programmation orienté objet en java, et aussi notre connaissance sur ce qu'on a trouvé concernant java fx. L'objectif c'est de concevoir et programmer d'une application, il nous a donné la possibilité de maîtriser et découvrir une nouvelle approche de la programmation.

## ***BIBLIOGRAPHIE***

<https://openjfx.io/>

<https://gluonhq.com/products/javafx/>

<https://www.oracle.com/fr/java/technologies/javase/javafx-overview.html>

**Fin**