

ALOUANI Fadi

BTS SIO 1

SLAM

B2

19/01/26

Programmation TD2 : ArrayList

Sommaire

<u>Introduction</u>	1
<u>Question :1</u>	2
<u>Question :2</u>	2
<u>Question :3</u>	3
<u>Question :4</u>	3
<u>Question :5</u>	4
<u>Question :6</u>	4
<u>Question :7</u>	4
<u>Code :</u>	5
<u>Livre.java</u>	5
<u>Main.java</u>	6
<u>Acceuil.java</u>	7
<u>Conclusion</u>	7

Introduction

Ce TP a pour objectif de se familiariser avec l'utilisation de l'ArrayList en Java à travers la gestion d'une classe Livre. Il permet de pratiquer la programmation orientée objet, la manipulation de listes dynamiques et l'interaction avec une interface graphique simple.

Question :1

Dans le programme principal créer une ArrayList de livre et ajouter le livre 1 et 2

```
import com.example.demo1.Livre;

import java.util.ArrayList;

public class Main {
    public static void main(String[] args) {

        ArrayList<Livre> listeLivres = new ArrayList<>();

        Livre livre1 = new Livre( number: "1111",  java: "Java",  s: "Auteur 1",  i: 10);
        Livre livre2 = new Livre( number: "2222",  java: "PHP",  s: "Auteur 2",  i: 12);

        listeLivres.add(livre1);
        listeLivres.add(livre2);
    }
}
```

Question :2

Ajouter 2 autres livres

```
Livre livre3 = new Livre( number: "3333",  java: "Python",  s: "Auteur 3",  i: 15);
Livre livre4 = new Livre( number: "4444",  java: "C++",  s: "Auteur 4",  i: 20);

listeLivres.add(livre3);
listeLivres.add(livre4);
```

Question :3

Afficher l'ensemble des livres(en utilisant la méthode afficher de la classe livre)

```
        livre4.Afficher();
        for (Livre livre : listeLivres) {
            System.out.println(livre);
        }
    }
```

```
C:\Users\eleve\.jdks\openjdk-25.0.1\bin\java
Titre: C++, Auteur: Auteur 4, Prix: 20€ Livre
Livre@e00cc92
```

Question :4

Créée une application Windows appeler accueil

```
Livre.java    Main.java    acceuil.java x
public class acceuil {
}
```

Question :5

Dans la classe Accueil créer un attribut privé Arrayliste de livre

```
import java.util.ArrayList;
ArrayList<Livre> listeLivres = new ArrayList<>();
```

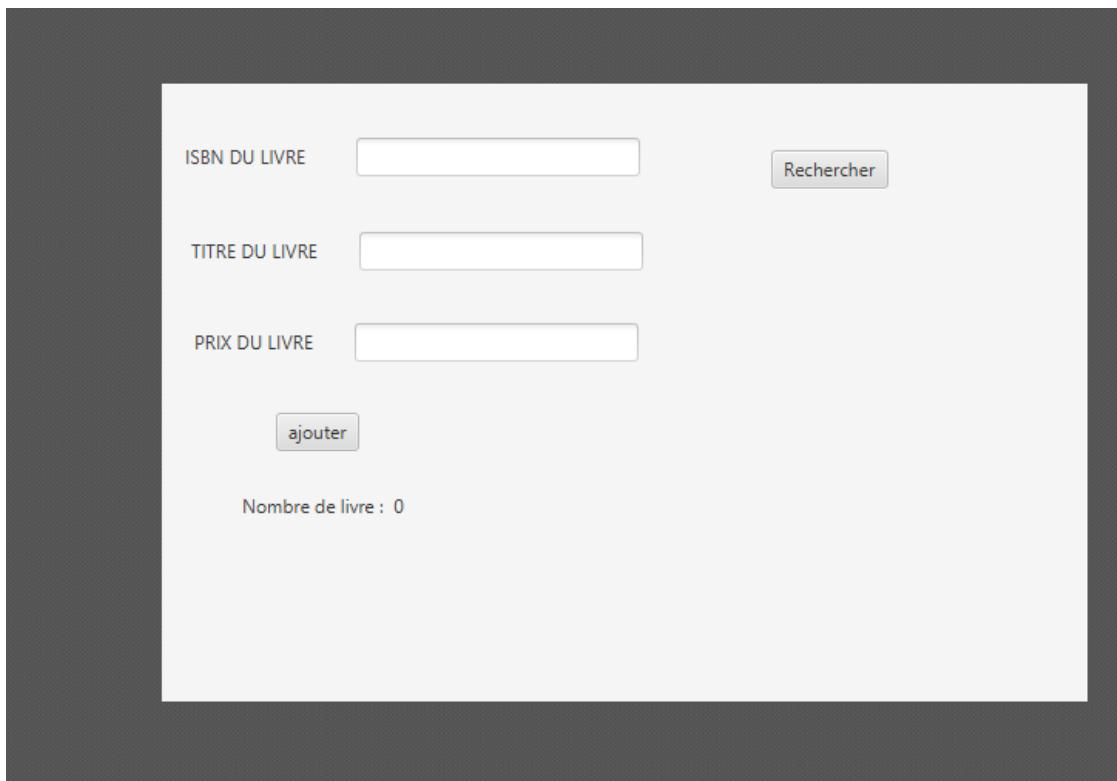
Question :6

Reprendre les 4 livres du main et les créer au lancement de l' application

Question :7

Refaire la fenêtre indiquer sur le TP

Interface graphique faite sur scenebuilder

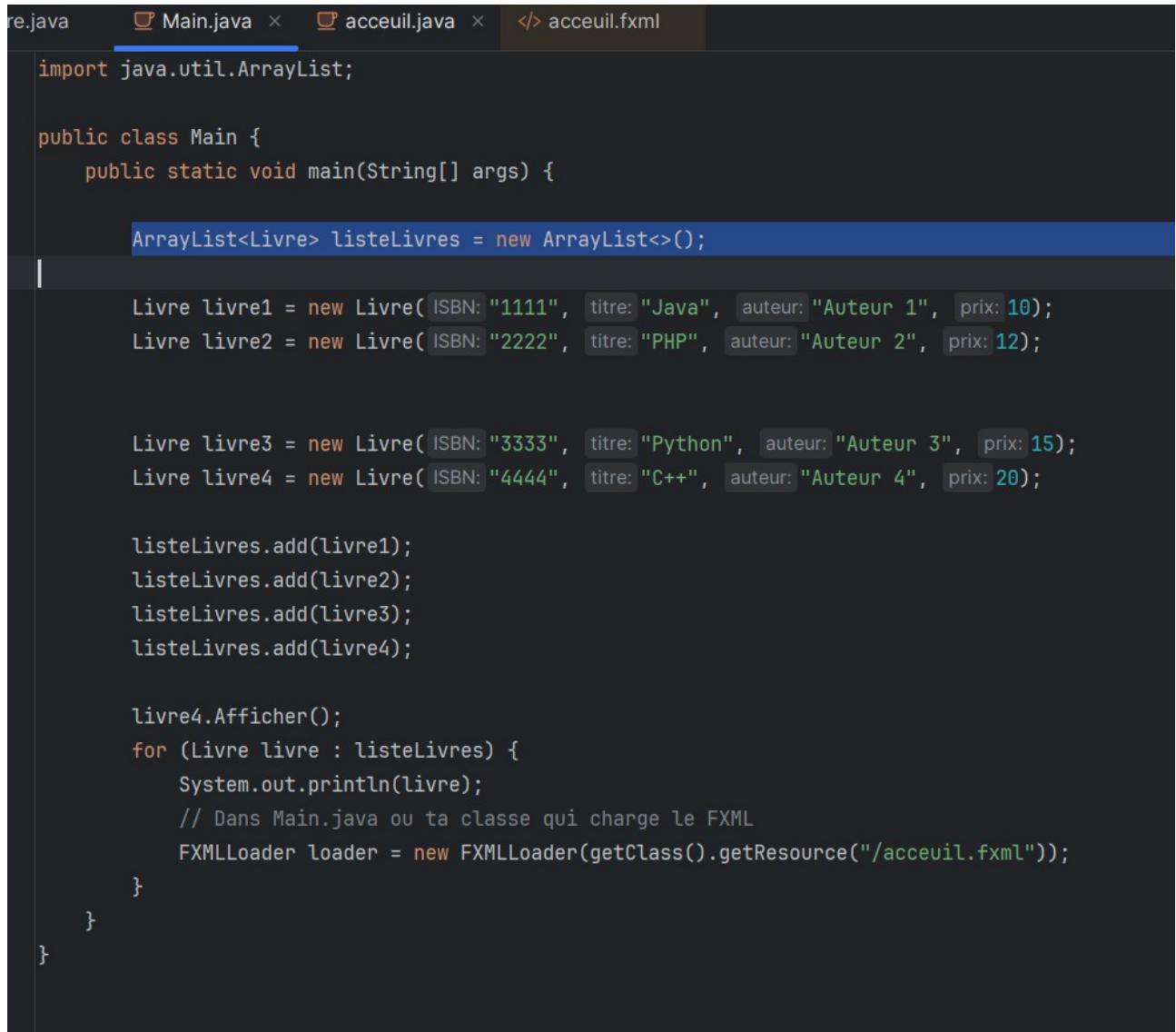


Code :

Livre.java

```
1 public class Livre {
2     private String ISBN, titre, Auteur;
3     private int prix;
4     //Début des GET et SET
5     >     public String getTitre() { return titre; }
6     >     public void setTitre(String titre) { this.titre = titre; }
7     >     public String getAuteur() { return Auteur; }
8     >     public void setAuteur(String auteur) { Auteur = auteur; }
9     >     public int getPrix() { return prix; }
10    >    public void setPrix(int prix) { this.prix = prix; }
11    //Constructeur du livre
12    public Livre(String ISBN, String titre, String auteur, int prix) {
13        this.ISBN=ISBN;
14        this.titre = titre;
15        Auteur = auteur;
16        this.prix = prix;
17
18    }
19    //Fonction d'affichage
20    public void Afficher(){
21        System.out.print("Titre: "+titre+",");
22        System.out.print("Auteur: "+Auteur+",");
23        System.out.print("Prix: "+prix+"€");
24    }
25    >     public String getISBN() { return ISBN; }
26    >     public void setISBN(String iISBN) { ISBN = iISBN; }
27
28
29
30
31
32
33
34
35
36
37    >     public String getISBN() { return ISBN; }
38    >     public void setISBN(String iISBN) { ISBN = iISBN; }
39
40
41
42
43
44 }
```

Main.java



The screenshot shows a Java code editor with four tabs at the top: 'Livre.java' (selected), 'Main.java' (underlined), 'acceuil.java', and 'acceuil.fxml'. The 'Main.java' tab is active, displaying the following code:

```
import java.util.ArrayList;

public class Main {
    public static void main(String[] args) {

        ArrayList<Livre> listeLivres = new ArrayList<>();

        Livre livre1 = new Livre( ISBN: "1111", titre: "Java", auteur: "Auteur 1", prix: 10);
        Livre livre2 = new Livre( ISBN: "2222", titre: "PHP", auteur: "Auteur 2", prix: 12);

        Livre livre3 = new Livre( ISBN: "3333", titre: "Python", auteur: "Auteur 3", prix: 15);
        Livre livre4 = new Livre( ISBN: "4444", titre: "C++", auteur: "Auteur 4", prix: 20);

        listeLivres.add(livre1);
        listeLivres.add(livre2);
        listeLivres.add(livre3);
        listeLivres.add(livre4);

        livre4.Afficher();
        for (Livre livre : listeLivres) {
            System.out.println(livre);
            // Dans Main.java ou ta classe qui charge le FXML
            FXMLLoader loader = new FXMLLoader(getClass().getResource("/acceuil.fxml"));
        }
    }
}
```

Acceuil.java

```
public class acceuil {  
    import java.util.ArrayList;  
    ArrayList<Livre> listeLivres = new ArrayList<>();  
}  
  
package ton.package; // le même que dans le projet  
  
import javafx.fxml.FXML;  
import javafx.scene.control.Button; // etc.  
  
public class acceuil {  
  
    @FXML  
    private Button monBouton; // exemple  
  
    @FXML  
    private void initialize() {  
        // ton code  
    }  
}
```

Conclusion

À la fin de ce travail, on comprend mieux comment créer, stocker et afficher des objets dans une ArrayList, ainsi que relier la logique métier à une interface utilisateur. Ce TD constitue une base importante pour développer des applications Java plus complètes et structurées.