

NFA035 – Bibliothèques et patterns : TP n°1 sur les Collections

V. Aponte

March 13, 2016

Exercice 1: travail avec ArrayList

L'objectif de cet exercice est de tester les principales opérations de la classe ArrayList. Pour cela, commencez par aller sur la page qui décrit la javadoc des ArrayList (site oracle). avec une méthode main.

Question 1

Dans une nouvelle classe, définissez une méthode statique `afficheListeSt (ArrayList<String> l)` permettant d'afficher un ArrayList de String. Dans le main, créez une ArrayList de String (n'oubliez pas d'importer `java.util.*`):

```
ArrayList<String> ls = new ArrayList<String>();
```

Vous allez effectuer les opérations suivantes sur cette liste.

1. Ajoutez successivement les chaînes "1", "2", "3", "4", "5" et "6" dans la liste. Affichez la liste avec la méthode `afficheListeSt`.
2. Remplacez l'élément de position 4 par la chaîne "remplacement".
3. Ajoutez dans la position 3 la chaîne "ajout".
4. Enlevez l'élément de position 2.
5. Affichez la liste après tous ces changements. Que constatez vous?
6. Que se passe-t-il si vous essayez d'enlever l'élément de position 20, ou d'ajouter un élément à la position 30?

Question 2

Ajoutez dans votre programme main le code suivant:

```
// Question 2
boolean b = ls.contains("4");
System.out.println("Contient_4?_" + b);

int index = ls.indexOf("4");
if(index == -1)
    System.out.println("Ne_contient_pas_4");
else
    System.out.println("Contient_4_a_la_position_" + index);
ls.add("1");
afficheListeSt(ls);

int lastIndex = ls.lastIndexOf("1");
```

```
if(lastIndex == -1)
    System.out.println("Ne_contient_pas_1");
else
    System.out.println("Derniere_occurrence_de_1:" + lastIndex);
```

Expliquez le comportement de ce code.

Exercice 2: livre de recettes (partie 1)

Dans cet exercice nous allons démarrer la programmation d'une application simplifiée de gestion d'un livre de recettes. Nous reprendrons ce code pendant plusieurs Tps afin de le compléter. N'oubliez pas de commenter votre code au format javadoc afin de pouvoir tester ces classes plus tard. Cet exercice doit être réalisé dans sa totalité au fil des Tps: si vous ne terminez pas aujourd'hui, continuez le avant d'entamer la suite au prochain tp.

Contraintes d'implantation

Pour ce Tp vous utiliserez un ArrayList afin de maintenir la liste des recettes courantes du livre. Lors des Tps suivants il vous sera demandé de changer la structure de données qui permet de stocker les recettes du livre. Faites donc particulièrement attention à écrire votre code de manière à ce qu'il soit facile à faire évoluer.

Question 1

Dans un premier temps, l'application doit offrir des opérations pour ajouter et supprimer des recettes du livre, pour donner une nouvelle note (de 1 à 5) à une recette, pour afficher une recette avec sa note courante et pour afficher le contenu de la liste de recettes courante (dans l'ordre de leur ajout). Vous définirez au moins deux classes: Recette et LivreRecettes.

Question 2

Ecrivez un programme main qui utilise vos classes. Lors du prochain tp vous écrirez la boucle qui propose un menu d'opérations et vous ajouterez également les traitements d'exceptions nécessaires pour gérer les erreurs.

Question 3

Vous devez maintenant faire en sorte que l'on puisse trier les recettes du livre:

1. par ordre des notes via un Comparator. Testez votre solution.
2. Modifiez maintenant la classe Recette de sorte qu'elle implante Comparable pour l'ordre alphabétique des recettes. Testez cet ordre.
3. Ajoutez des méthodes dans la classe LivreRecettes permettant d'afficher les recettes selon les deux ordres implantés.