

TD n°10 – Tableaux, collections, tests unitaires

Ce TD/TP a pour but d'utiliser les services de la classe Arrays sur les tableaux, d'utiliser des collections génériques de type ArrayList<T>, et dictionnaire HashMap. Enfin, la mise en œuvre de l'automatisation des tests unitaires gérée avec JUnit sous Eclipse est abordée.

Services de la classe Arrays

<https://docs.oracle.com/javase/8/docs/api/java/util/Arrays.html>

La classe Arrays du package java.util offre des services pour les tableaux (tri, recherche, remplissage, égalité, ...) La liste des méthodes définies dans la classe Arrays sont static et s'appellent donc en utilisant le nom de la classe suivi de celui de la méthode : Arrays.sort(Object[] a).

Soient les tableaux instanciés suivants :

```
int[] T1 = new int[10] ;
String[] T2 = new String[7] ;
int[] T3 = {0, 6, 2, 4, 3} ;
String[] T4 = {« bleu », « rouge », « blanc », « vert », « mauve », « indigo »};
```

Consulter les nombreuses méthodes disponibles dans la classe Arrays et trouver les méthodes permettant de faire les actions suivantes :

- Remplir T1 avec l'entier 5 dans toutes les cases.
- Remplir T2 avec la chaîne de caractères « bleu » dans les cases d'indices 1 et 2.
- Afficher les tableaux
- Trier T3 dans l'ordre croissant
- Trier T4 dans l'ordre alphabétique
- Vérifier si T1 et T3 sont égaux
- Copier les 5 premiers éléments de T4 dans un nouveau tableau T5
- Dupliquer T4 dans un tableau T6 et vérifier s'ils sont égaux
- Transformer T4 en List<String> et afficher la liste obtenue

Créer une classe avec une méthode main afin de tester les méthodes demandées ci-dessus.

Comparaison d'objets

Une classe peut spécifier un **ordre naturel** en implantant l'interface Comparable<T>

```
import java.util.*;

public interface Comparable<T> {
    int compareTo (T t);
}
```

La fonction *compareTo* renvoie :

- < 0 si this est *inférieur* à T
- == 0 si this est *égal* à T
- > 0 si this est *supérieur* à T

T doit être la classe spécifiant l'ordre. L'ordre est défini dans la classe T de manière explicite.

1. Récupérer les classes fournies en annexe et les compléter.
2. Consulter la classe Collections du package java.util (<https://docs.oracle.com/javase/8/docs/api/java/util/Collections.html>) et regarder les méthodes disponibles. Quelle est la différence avec la classe Arrays ?
3. Afficher les résultats de la méthode compareTo entre les 3 personnes.
4. Commenter la partie « implements Comparable<Personne> » dans la ligne d'en-tête de la classe Personne ainsi que la méthode compareTo. Que se passe-t-il ? Où se trouve l'erreur ? Expliquer en se référant à la documentation.
5. On souhaite maintenant pouvoir trier une collection de personnes suivant leur âge.

L'interface Comparator<T> permet de spécifier **un ordre externe** (temporaire)

```
import java.util.*;

public interface Comparator<T> {
    int compare(T t1, T t2);
}
```

La valeur de retour de compare suit les mêmes règles que compareTo, mais l'ordre externe entre les 2 objets en paramètre est un ordre valable juste à un moment donné (rien de naturel et évident).

Cette méthode s'utilise en paramètre de la méthode Collections.sort afin de trier temporairement une collection en fonction d'un Comparator<T>.

- a. Ajouter un attribut int age. A quel niveau de la hiérarchie se trouve-t-il ?
- b. Ajouter les accesseurs correspondants
- c. Modifier le main afin de trier la liste avec ce nouveau comparateur

Dictionnaire

Développer une classe abstraite Traducteur, permettant de traduire des mots français en anglais.

Cette classe dispose d'un attribut de classe *m_dico* de type HashMap<String, String>.

Ses méthodes et services sont les suivants :

- **init** pour initialiser l'attribut avec un dictionnaire passé en paramètre
- **getDico** Accesseur de consultation qui renvoie l'attribut
- **traduire** Méthode qui retourne la traduction anglaise d'un mot français passé en paramètre
- **inverse** Méthode qui retourne la traduction française d'un mot anglais passé en paramètre

Prise en main de JUnit sous Eclipse

Réaliser pas à pas le tutoriel disponible à l'adresse suivante :

<http://www.junit.fr/2011/11/20/tutoriel-eclipse-junit-mon-premier-test-automatique/>

Créer des nouvelles classes de tests pour tester :

- La classe String comme dans l'exemple du cours
- La classe Point précédemment utilisée dans les TDs