

## 2. Feuille de TD

semaine 14 :

### Simple

On vous demande d'implémenter une classe générique **Simple** possédant un seul attribut, deux constructeurs (l'un sans paramètre, l'autre avec un paramètre du type de l'attribut), un getter et un setter. Vous testerez cette classe dans un programme principal avec un **Integer** (12), un **String** ("toto") et un **Double** (5.5). Est-il possible de déclarer un Simple de Simple d'entier ? comment ? et un Simple de Simple d'un Couple de chaînes ?

### Rappels

L'interface Comparable<T> permet de définir des objets 'naturellement comparables', elle est utilisée par la méthode sort de Collections :

```
public static <T extends Comparable<T>> void sort(List<T extends Comparable<T>>)
```

Pour être Comparable<T> il faut implémenter :

```
public int compareTo(T)
```

- Écrivez une classe CoupleEntiers qui soit naturellement triable par premier élément croissant.
- Écrivez une classe CoupleEntiers qui soit naturellement triable par premier élément croissant et dans le cas d'égalité, compare par second élément.
- Écrivez une classe CoupleDouble qui soit naturellement triable par premier élément croissant.
- Définissez une exécutable.

### Couple générique

- Écrivez une classe générique Couple<T extends Comparable<? super T>> qui soit naturellement triable par premier élément croissant. *Remarque* faites attention au profil de la méthode compareTo.
- Remarquez qu'il suffirait pour la classe T d'être comparable avec n'importe lequel de ses ancêtres dans la hiérarchie, c'est-à-dire que si par exemple T étend une classe qui est Comparable <un des parents de T> alors T peut être comparé avec T. On peut alors modifier la classe pour avoir le profil : Couple<T extends Comparable<? super T>>.
- Tester cette classe dans un exécutable, sur des entiers et des doubles ainsi que sur des couples d'entiers ou de doubles.

### Ajoutable

- Écrivez une interface Ajoutable<T> contenant une méthode T ajoute(T x). Cette méthode, lorsqu'elle sera implémentée, correspondra à la notion de +. Par exemple si on a une classe Entier implémentant Ajoutable<Entier>, cinq.ajoute(trois) renverrait huit.
- Écrivez une classe Entier implémentant Ajoutable.
- Écrivez une classe Ajoutables (avec un s) contenant une méthode somme prenant en entrée une liste d'Ajoutable et renvoyant la somme de ces Ajoutables. Essayez de trouver le profil de somme.
- Utilisez la méthode précédente pour faire la somme des éléments d'un tableau d'Entiers, dans un exécutable.
- Utilisez la méthode précédente pour pouvoir faire la concaténation de chaînes de caractères. Que faut-il faire ?