

Java DUT 1 Feuille TP6
Université Paris-Est Marne-la-Vallée

Exercice 1.—

On reprend la classe `Book` du TP3 qui permet de construire des livres qui ont un titre (`title`) et un auteur (`author`).

On souhaite modéliser la gestion d'un catalogue de livres. Un livre peut figurer ou non dans le catalogue et s'il figure il peut figurer en plusieurs exemplaires. Le libraire doit pouvoir ajouter et enlever **un exemplaire** d'un livre du catalogue et aussi consulter le nombre d'exemplaires présents en stock d'un livre donné.

- a) Écrire une interface `Catalog` comprenant des méthodes

```
– void add(Book b)
– void remove(Book b)
– int getNumber(Book b)
```

- b) Écrire dans la classe `Book` une méthode `public boolean equals(Object o)` renvoyant `true` si deux `Book` ont les même champs et `false` sinon. Écrire une méthode `public int hashCode()` renvoyant un entier. Deux `Book` égaux par `equals` devront avoir le même `hashCode`.

- c) Écrire une classe `ImplementationCatalog` implémentant l'interface `Catalog`. Le `Catalog` sera représenté par une `Map`, un objet que l'on pourra appeler `map` de la classe `HashMap<Book, Integer>`.

```
public class ImplementationCatalog implements Catalog {
    private final Map<Book, Integer> map;
```

Une telle `Map` est une table de correspondance indiquant pour chaque `Book` son nombre d'exemplaires.

- Écrire un constructeur dans `ImplementationCatalog`.
 - Écrire la méthode `void add(Book b)` qui permet d'ajouter un exemplaire d'un livre. Si le livre n'était pas dans la `Map`, il sera ajouté en un exemplaire.
 - Écrire la méthode `int getNumber(Book b)` qui renvoie le nombre d'exemplaire d'un livre. On renverra 0 si le livre n'est pas présent dans le catalogue.
 - Écrire la méthode `void remove(Book b)` qui permet de retirer un exemplaire d'un livre. Si le livre apparaît avec un nombre d'exemplaire 0, il sera retiré de la `Map`.
 - Écrire une méthode `toString` dans `ImplementationCatalog`.
- d) Écrire une classe `CatalalogTest` avec un méthode `main` et tester vos classes.
- e) A quel moment sont utilisées les méthodes `equals` et `hashCode` de `Book` ? Pourquoi est-il important que les champs de `Book` soient `final` ?

- f) Ajouter une méthode `size` renvoyant la taille d'un catalogue qui est le nombre total d'exemplaires présents dans le stock. Dans quelles classes faut-il la mettre ?

Exercice 2.—

Ecrire une classe `WordCount` contenant une méthode `main` qui prend en argument une suite de mots entrés au clavier et retourne la liste des mots avec leur nombre d'occurrences.

Par exemple, si on lance

```
java WordCount toto titi toto tutu tata titi titi
```

la sortie obtenue sera

```
tata 1
titi 3
tutu 1
toto 2
```

Comment modifier le programme pour que la suite des mots avec leur nombre d'apparition soit en ordre alphabétique ?