

Dans ce TP, on utilise la classe java `ArrayList` qui permet de construire des tableaux de taille variable. Pour pouvoir l'utiliser, il faut importer le package correspondant grâce à la commande `import java.util.ArrayList;`

Un tableau de taille variable de nom `tab` et contenant des `String` se déclarera par : `ArrayList<String> tab;` et s'initialisera avec la commande `tab = new ArrayList<String>();`

Le but de ce TP est de manipuler une médiathèque contenant un certain nombre de médias. Un média possède un titre, un support (exemple : livre, dvd, etc), ainsi qu'un nombre d'exemplaires.

Exercice 1 [Media & Mediatheque]

1. Créer une classe `Media` contenant
 - (a) Trois attributs :


```
private final String titre;
private final String support;
private int nbExemplaires;
```
 - (b) Les trois «getteurs» appropriés.
 - (c) Un «setteur» pour `nbExemplaires`.
 - (d) Un constructeur `Media(String titre, String support)` dont les arguments correspondent aux deux premiers attributs et qui met `nbExemplaires=1`.
 - (e) Une méthode `void affiche()` qui affiche


```
Poly de TO2 (Livre) x2
si support="Livre", titre="Poly de TO2" et nbExemplaires=2.
```
 - (f) Une méthode `boolean estEgalA(Media el)` qui renvoie `true` si `this` et `el` ont des `support` et `titre` identiques.
2. Dans une classe `Principal` créer une méthode `main` pour tester tout ceci. On fera tout particulièrement attention au test de `estEgalA`.
3. Créer une classe `Mediatheque` contenant
 - (a) Un attribut `ArrayList<Media> contenu`.
 - (b) Un constructeur `Mediatheque()` qui initialise l'attribut à un tableau vide.
 - (c) Une méthode `void ajoute(Media el)` qui ajoute `el` dans `contenu` s'il n'y est pas déjà; s'il y est déjà, la méthode doit augmenter la valeur de son attribut `nbExemplaires`.
 - (d) Une méthode `void affiche()` qui affiche tous les `Media` de la collection, un par ligne.
4. Tester dans le `main` de `Principal` avec le code suivante

```
Mediatheque mesMedias = new Mediatheque();
mesMedias.ajoute(new Media("Poly de IP2","Poly"));
mesMedias.ajoute(new Media("Poly de IP2","Poly"));
mesMedias.ajoute(new Media("Poly de IP2","Poly"));
mesMedias.ajoute(new Media("Poly de IF1","Poly"));
mesMedias.ajoute(new Media("Les Miserables", "Livre"));
mesMedias.ajoute(new Media("Les Miserables", "Dvd"));
mesMedias.ajoute(new Media("Les Miserables", "Livre"));
mesMedias.affiche();
```

qui doit donner l'affichage :

```
Poly de IP2 (Poly) x3
Poly de IF1 (Poly) x1
Les Miserables (Livre) x2
Les Miserables (Dvd) x1
```

Exercice 2 [Copie]

1. Ajouter dans la classe `Media` un constructeur `Media(Media el)` qui produit une copie de l'argument.
2. En utilisant le constructeur précédent, écrire dans la classe `Mediatheque` le constructeur `Mediatheque(Mediatheque collec)` qui produit une copie de `collec`.

Dans la suite du TP on supposera qu'un média de `titre` et de `support` fixés n'apparaît qu'une fois dans la `Mediatheque`.

Exercice 3 [Inclusion & Égalité] Ajouter dans la classe `Mediatheque` les deux méthodes suivantes.

1. `boolean estContenuDans(Mediatheque collec)`, qui renvoie `true` si `this` est contenu dans `collec`.
Attention, pour un `Media` qui est dans les deux, il faut vérifier que la quantité dans `this` est inférieure ou égale à la quantité dans `collec`.
2. `boolean estEgaleA(Mediatheque collec)`, qui renvoie `true` si `collec` et `this` ont les mêmes médias en même quantité.
Indice : utiliser la méthode précédente.

Exercice 4 [Filtre]

1. Ajouter dans la classe `Mediatheque` une méthode `Mediatheque filtreParTitre(String titre)` qui crée une nouvelle `Mediatheque` contenant uniquement les `Media` de `this` dont le titre est identique à celui donné en argument.
Attention à copier les `Media` et à ne pas seulement copier leur référence.
Tester cette méthode, par exemple en filtrant la `Mediatheque mesMedias` (définie précédemment) par le titre "Les Miserables".

Exercice 5 [Somme] (*facultatif*)

Ajouter dans la classe `Mediatheque` la méthode et le constructeur suivants :

1. `void ajouteMediatheque(Mediatheque c)`, qui ajoute tous les `Media` de `c` dans `this`.
Tester par exemple en ajoutant `mesMedias` à elle-même.
2. `Mediatheque(ArrayList<Mediatheque> tabDeCollec)`, qui construit une `Mediatheque` contenant la réunion des `Mediatheque` données en argument.

Exercice 6 [Doublon] (*facultatif*)

1. Ajouter dans la classe `Mediatheque` une méthode `Mediatheque retireDoublon()` qui ne laisse dans `this` qu'un seul exemplaire de chaque `Media`, et met tous les doublons dans une nouvelle `Mediatheque` qui est renvoyée.
Tester cette méthode avec `mesMedias` doit donner les affichages suivants :

```
[mesMedias.retireDoublons().affiche()]      [mesMedias.affiche()]
Poly de T02 (Poly) x2                       Poly de T02 (Poly) x1
Les Miserables (Livre) x1                   Poly de IF1 (Poly) x1
                                             Les Miserables (Livre) x1
                                             Les Miserables (Dvd) x1
```