

## Programmation Orientée Objets

### Héritage

E. Hyon L. Mesnager

2022-2023

Ce premier TD a pour but la mise en œuvre des concepts liés à l'héritage. Ils contiennent 3 Exercices.

#### Exercice 1 : Héritage : les livres

On suppose 3 types d'ouvrages. Les livres caractérisés par : un titre, un auteur, un nombre de pages et un éditeur ; les oeuvres qui, en plus des champs précédents, ont : une critique (note entre 0 et 5) et une recommandation (booléen) et enfin les chefs d'oeuvres qui, en plus des champs de livres et des oeuvres, sont caractérisés par : une popularité (pourcentage).

##### Question 1

Écrire les trois classes et les constructeurs associés (pour des besoins de pédagogie les champs de livres auront des attributs de visibilité différents).

N.B. pour faciliter les comparaisons ultérieures tous les champs de type `String` seront encodés en minuscules.

##### Question 2

Nous allons raffiner les constructeurs.

On suppose que on va avoir un constructeur qui est spécifique à l'éditeur *p.u.m.n* (Presses Universitaires de la Miage Nanterre). Et on va avoir un second constructeur spécifique lui à l'auteur *Mélanie Notlong* dont les livres font tous 501 pages (les autres champs sont libres).

Pour les classes filles tous les ouvrages des *p.u.m.n* qui sont des oeuvres ont une belle critique (note de 5) tandis que les livres de *Notlong* qui sont des chefs d'oeuvre ont une forte popularité (0.8).

##### Question 3

Redéfinissez la méthode `toString()` pour chacune des classes.

##### Question 4

Tout livre peut être lu et la méthode lecture renvoie un temps de lecture proportionnel au nombre de pages pour les livres. Ce nombre est un entier. Le temps de lecture est proportionnel d'un facteur `p` interne à la méthode `lecture` dont la valeur est laissée libre au programmeur. La lecture d'une oeuvre prend le double d'un livre et la lecture d'un chef d'oeuvre prend  $3/2$  du temps d'une lecture d'une oeuvre.

### Question 5

On veut faire afficher des messages dans des méthodes de classe.

Un message serait “*Oh un livre*” pour les livres, “*J’aime les classiques*” pour les oeuvres et “*Ah que c’est beau*” pour les chefs d’oeuvres. Ecrivez la méthode statique `message` qui affiche ce message pour chacune des classes.

Ecrivez une méthode de classe `passion` qui affiche “*la lecture c’est mon dada*” pour toutes les classes.

### Question 6

Toute oeuvre bénéficie d’une méthode `like` qui, quand on l’applique, permet de donner une nouvelle valeur à la recommandation. Cette valeur est aléatoire et dépend d’une probabilité spécifique (60% de chance de donner vrai pour les oeuvres et 70% pour les chefs d’oeuvre). La classe `Random` de `java.util` pourra vous être utile.

### Question 7

Ecrivez la classe `main` en enregistrant les livres suivants <sup>1</sup> :

Denis-Henri Lévy, *Barbés Vertigo*, Le découvert, 100 pages

Marc Levis, *Et si c’était niais ?*, Gallimatias, 245 pages

Anna Galvauda, *Quelqu’un m’attend, c’est tout*, Minuit, 10 pages.

Mélanie Notlong, *hygiène du tube (et tout le tremblement)* chez gallimatias

Mélanie Notlong *Cosmétique de l’ennui* chez Soule

Et au p.u.m.n : C. Alanoy / *Java Exercices à savoir faire pour passer en master*, 6000000 pages.

Sachant que les livre *Quelqu’un m’attend, c’est tout* et *hygiène du tube (et tout le tremblement)* sont des oeuvres avec une recommandation fausse et une critique de un. Sachant également que *Et si c’était niais ?* et *Java....* sont des chefs d’oeuvre avec une popularité qui sera remise à jour par la méthode `like`. Affichez les caractéristiques de chaque livre, leur temps de lecture et également les messages de classe et le message de passion.

### Question 8

Ecrivez la méthode `afficheTitre()` dans `Livre`. Placez tous vos livres dans un tableau dans la méthode `main`.

Est-il possible d’ajouter des Oeuvres et des chef d’Oeuvres à ce tableau et pourquoi ?

Affichez les titres du tableau avec un parcours sous la forme d’un `for each` qui en java est :

```
for(Livre c : tableau).
```

Doit-on redéfinir la méthode `afficheTitre()` pour les oeuvres et les chef d’oeuvres ?

Ecrivez la méthode `afficheRecommandation()` dans `Oeuvre`. Est-il possible d’afficher les recommandations ?

## Exercice 2 : Y a marqué la poste ici

On suppose qu’un guichet doit gérer plusieurs types de courrier recommandé. Les lettres recommandées simples, les lettres recommandées avec accusé de réception, les lettres recommandées avec accusé de réception et service de suivi personnalisé. Les envois sont gérés par des guichets qui peuvent être de plusieurs types (demi guichet, rapide, lent), un guichet permet de déposer un courrier ou de le récupérer.

Le courrier peut être dans les différents états suivants :

1. En instance de tamponnement
2. En acheminement
3. Déposé chez le destinataire
4. Accepté par le destinataire

Deux types d’utilisateurs qui peuvent accéder aux données : les personnels de la poste et les particuliers dont le courrier est du bon type.

1. les pastiches sont tirés du livre : la rentrée littéraire assassinée de P. Fioretto

**Question 1**

Faites une modélisation des différentes fonctionnalités à mettre en oeuvre.

**Question 2**

Programmez cela.

## 1 Exercices supplémentaires

**Exercice 3 : A vous de jouer**

On a deux classes une classe Débit de Boisson `debitBoisson` et une classe `Bar` qui est un raffinement du Débit de Boisson. Un débit de Boisson a un attribut nom de société, un attribut C.A. (Chiffre d'affaire) et un attribut licence. Le bar a en plus un attribut surface et un attribut nom du bar.

**Question 1**

Ecrivez les deux classes sachant que :

- La classe débit de Boisson a 3 constructeurs. Chaque constructeur finit par l’affichage d’une phrase qui lui est propre. On a un constructeur dont tous les attributs ont une valeur en paramètre. Un constructeur dont le nom de société est fixé ("bognatInternational") et le troisième dont le niveau de licence est fixé.
- La classe bar a 4 constructeurs. Dans l’un la surface est fixe et la licence est de niveau 4 et tous les autres attributs sont en paramètres du constructeur. Dans le second, tous les attributs sont en paramètres du constructeur. Dans le troisième la surface et le nom de société ("bognatInternational") sont fixés. Enfin, le quatrième fait appel au premier constructeur.

Par ailleurs les attributs de visibilité sont ceux par défaut.

**Question 2**

Ecrivez la méthode `toString()`.

**Question 3**

Ecrire la méthode `tropStyle` qui affiche la rapport CA/ licence pour les débits de Boisson et le rapport CA/surface pour les bars.

**Question 4**

Créer une classe `Main` et faites afficher les caractéristiques des débits. Vous créerez un bar et un débit de boisson.

**Question 5**

Modifiez votre attribut de visibilité de l’attribut C.A. en `private`. Quels impact cela a sur votre méthode `tropStyle`? Comment la modifier sans faire appel à des `getter`

**Question 6**

Comment modifier `tropStyle` en faisant appel à des `getter`?