

## Évaluation POO & Java

avril 2021  
responsable : M.Munier

### Projet - Gestion Médiathèque

**Remarque :** Étant donné la période de confinement pour lutter contre l'épidémie de Covid-19, l'évaluation pratique du module M2207 se fera cette année au travers d'un mini projet. La description des classes vous est fournie ; il ne vous reste plus qu'à coder ces classes en Java.

★ ★ ★

## 1 Présentation du projet

Le thème de ce mini-projet est la conception d'une base de données permettant la gestion des œuvres d'une médiathèque. Ces œuvres peuvent être de différents types : livre, vidéo, musique, etc. Nous allons pour cela développer plusieurs classes. La première, la classe **Oeuvre**, nous permettra de regrouper tous les attributs qui sont communs aux différents types d'œuvres. Les classes **Livre**, **Video** et **Musique** hériteront de la classe **Oeuvre** pour ajouter des attributs qui leur sont spécifiques. Voici pour chacune des classes les attributs qu'elles devront stocker :

- **Oeuvre** → **reference** (code d'enregistrement à la médiathèque), **auteur**, **titre**, **annee** (de publication)
- **Livre** → **isbn** (*International Standard Book Number*), **editeur**, **pages** (nombre de pages)
- **Video** → **format** (MPEG, MP4, DivX, ...), **duree**
- **Musique** → **support** (CD, DVD, MP3, vinyle), **pistes** (nombre de chansons ou pistes musicales)

## 2 Travail à réaliser

Dans le cadre de ce mini-projet, votre travail consiste à implémenter ces classes en Java, à les tester individuellement, puis à écrire un petit programme de test global. Ne perdez pas de temps avec une interface graphique quelconque ! Concentrez vous sur l'écriture des classes (avec les attributs, les constructeurs, les méthodes), l'algorithmique (code dans les méthodes) et l'utilisation de ces classes (instanciation d'objets, appels de méthodes).

NB : Les interactions avec l'utilisateur se feront en mode console, directement dans le terminal (avec des `System.out.println` et des `Scanner`).

1. En ce qui concerne la classe **Oeuvre** :
  - les attributs communs indiqués précédemment
  - les méthodes d'accès correspondantes
  - une méthode **afficher()** permettant... d'afficher le détail de cette œuvre

2. Nous aurons ensuite les différentes sous-classes **Livre**, **Video** et **Musique** :
  - ajouter les attributs nécessaires à chacune des sous-classes
  - ajouter les méthodes d'accès correspondantes
  - écrire les constructeurs adéquats (en faisant appel aux constructeurs de la classe mère)
  - redéfinir la méthode **afficher()**
3. Nous définirons ensuite la classe **Mediatheque** qui contiendra une **liste d'œuvres** (i.e. un tableau de références vers des instances de la classe **Oeuvre**) et fournira un certain nombre de méthodes pour :
  - ajouter une œuvre
  - afficher la liste de toutes les œuvres enregistrées
  - rechercher des œuvres à partir d'un mot-clé dans le titre, du nom de l'auteur, etc...
  - calculer le nombre d'œuvres pour une année donnée (année de publication)
4. Écrire un **programme principal** permettant de gérer une médiathèque :
  - gestion d'un petit menu (en mode console)
  - saisie des informations au clavier
  - instanciation d'objets à partir des différentes classes
  - appel des méthodes sur les objets créés
  - affichage des résultats

### 3 Restitution de vos travaux

- Vous développerez toutes les classes de ce projet dans un seul et unique répertoire dont le nom est **votre nom de famille en minuscules**. Pour la restitution de votre travail, vous créerez une archive (tar ou zip) de ce répertoire, le nom de cette archive étant lui aussi **votre nom de famille en minuscules**. Vous enverrez ensuite cette archive par mail à l'adresse [manuel.munier@gmail.com](mailto:manuel.munier@gmail.com).
- Il s'agit d'un travail individuel. Vous pouvez bien évidemment vous entraider, vous documenter sur Internet, etc. Mais en aucun cas je n'accepterai d'avoir le même projet en x exemplaires ! L'objectif d'un tel projet, avant toute chose, est de renforcer vos compétences en programmation. Inutile de tricher aujourd'hui en faisant faire votre projet par quelqu'un d'autre, car vous payerez les conséquences de vos actes demain... (cf. M2105, M4207C, projets tutorés, poursuites d'études, etc.)
- La date limite pour rendre votre projet est fixée au **mardi 27 avril 2021 à 12h**, date de réception du mail faisant foi.

★ ★ ★

#### Informations complémentaires :

- Afin de vous aider dans votre tâche, vous trouverez la documentation en ligne du Java SE 10 à l'URL suivante : <https://docs.oracle.com/javase/10/docs/api/index.html?overview-summary.html>.
- Pour mémoire, **String** est une classe. Pour comparer deux chaînes de caractères vous devrez donc utiliser la méthode **equals** et non pas faire une comparaison des références. Par exemple, si vous avez deux chaînes **chA** et **chB**, la comparaison se fait via l'instruction **chA.equals(chB)** qui retourne **true** ou **false**.
- De la même façon, pour savoir si une chaîne de caractères **chB** est contenue dans une chaîne **chA**, nous utiliserons l'instruction **chA.indexOf(chB)** qui retourne un entier indiquant la position (1<sup>er</sup> caractère) de **chB** à l'intérieur de **chA**. Si la chaîne **chB** n'est pas trouvée dans **chA**, alors la méthode **indexOf** renvoie la valeur **-1**.