

# Gestionnaire de Bibliothèque

## Description :

Ce projet consiste à développer un système de gestion de bibliothèque en utilisant les principes de la programmation orientée objet en Java. Le système devrait permettre aux utilisateurs d'effectuer des opérations telles que l'ajout, la suppression, la recherche et la gestion des livres dans une bibliothèque virtuelle.

## Fonctionnalités suggérées :

### Gestion des Livres :

- Ajouter un nouveau livre avec des détails tels que le titre, l'auteur, l'année de publication, le numéro ISBN, etc.
- Modifier un livre existant
- Supprimer un livre de la bibliothèque.
- Rechercher un livre par titre, auteur ou ISBN.

### Gestion des Emprunts :

- Enregistrer les emprunts et les retours de livres.
- Afficher les livres empruntés par un utilisateur donné.
- Limiter le nombre de livres qu'un utilisateur peut emprunter simultanément.

### Gestion des Utilisateurs :

- Enregistrer de nouveaux utilisateurs avec des informations telles que le nom, le numéro d'identification, etc.
- Vérifier l'éligibilité des utilisateurs à emprunter des livres en fonction de leurs cotisations. Un utilisateur ne pourra effectuer d'emprunt que s'il est à jour par rapport à ses cotisations (un booléen pourra être utilisé pour enregistré cet état).

### Interface Utilisateur :

- Développer une interface utilisateur conviviale pour interagir avec le système.
- Afficher les menus et les options pour permettre aux utilisateurs d'effectuer différentes opérations.

### Points supplémentaires :

- Implémenter des classes de livres spécialisées, telles que des romans, des essais, des livres audio, etc., qui héritent des fonctionnalités de la classe de base Livre. [MASTER]
- Utiliser des collections Java telles que ArrayList ou HashMap pour stocker les livres et les utilisateurs.
- Gérer les cas d'exception pour gérer les erreurs potentielles lors de l'interaction avec le système (par exemple, tentative d'emprunt d'un livre déjà emprunté). [MASTER]

- Ajouter des fonctionnalités avancées telles que la génération de rapports statistiques sur les livres les plus empruntés, les utilisateurs les plus actifs, etc. [MASTER]

### Livrables attendus :

- Code source Java bien documenté avec des commentaires expliquant la logique derrière chaque fonctionnalité [à mettre sur GitHub]
- Une vidéo de démonstration du système, mettant en évidence différentes fonctionnalités à travers une série de scénarios d'utilisation. [à mettre sur YouTube]

## Quelques indications sur la structure des classes

### Classe Livre :

- Attributs :
  - String titre
  - String auteur
  - int anneePublication
  - String ISBN
- Méthodes :
  - Constructeur pour initialiser les attributs.
  - Getters et setters pour accéder et modifier les attributs.
  - Méthode toString() pour afficher les détails du livre.

### Classe Utilisateur :

- Attributs :
  - String nom
  - int numeroIdentification
  - ArrayList<Livre> livresEmpruntes
- Méthodes :
  - Constructeur pour initialiser les attributs.
  - Méthode pour emprunter un livre.
  - Méthode pour retourner un livre.
  - Méthode pour afficher les livres empruntés par l'utilisateur.

### Classe Bibliothèque :

- Attributs :
  - ArrayList<Livre> listeLivres
  - HashMap<Utilisateur, ArrayList<Livre>> empruntsUtilisateurs
- Méthodes :
  - Méthode pour ajouter un livre à la bibliothèque.
  - Méthode pour supprimer un livre de la bibliothèque.
  - Méthode pour rechercher un livre par titre, auteur ou ISBN.
  - Méthode pour enregistrer l'emprunt d'un livre par un utilisateur.
  - Méthode pour enregistrer le retour d'un livre par un utilisateur.
  - Méthode pour vérifier l'éligibilité d'un utilisateur à emprunter un livre.

- Méthode pour afficher les statistiques de la bibliothèque (nombre total de livres, nombre d'exemplaires empruntés, etc.).

### Classe Main (Point d'entrée) :

Cette classe contiendra la méthode main() pour démarrer l'application. Elle initialisera la bibliothèque et fournira une interface utilisateur pour interagir avec les fonctionnalités du gestionnaire de bibliothèque.

### Classe LibraryException : [MASTER]

Cette classe gèrera les exceptions qui pourraient être levées lors de l'exécution du programme, telles que les livres non trouvés, les utilisateurs non trouvés, les emprunts non autorisés, etc.