

Vous devez implémenter une hiérarchie de classes pour gérer une bibliothèque. La classe abstraite Document sera la classe de base, et les classes Livre, Revue, et CD en hériteront pour définir leurs spécificités. Ensuite, vous devez implémenter la classe Bibliothèque, qui contiendra une liste de documents et permettra d'ajouter, supprimer et mettre à jour des documents.

1. Classe Document (classe abstraite)

La classe Document est une classe abstraite qui définit les attributs communs à tous les documents et les méthodes de base.

Attributs :

- `id (int)` : un identifiant unique pour chaque document.
- `titre (str)` : le titre du document.
- `auteur (str)` : l'auteur ou le créateur du document.
- `annee_publication (int)` : l'année de publication du document. Une exception doit être levée si l'année de publication est inférieure à 1990

Méthodes :

- `__init__(self, id, titre, auteur, annee_publication)` : constructeur pour initialiser les attributs.
- `afficher_details(self)` : méthode pour afficher les détails d'un document (à implémenter dans les classes filles).
- `modifier_details(self, titre=None, auteur=None, annee_publication=None)` : méthode permettant de modifier certains attributs du document. Les attributs peuvent être laissés à None s'ils ne doivent pas être modifiés.
- `__str__(self)` : méthode de représentation pour afficher les informations générales sur le document.

2. Classe Livre (hérite de Document)

La classe Livre représente un livre, un type de document spécifique.

Attributs :

- `isbn (str)` : le numéro ISBN du livre.
- `nombre_de_pages (int)` : le nombre de pages du livre.

Méthodes :

- `__init__(self, id, titre, auteur, annee_publication, isbn, nombre_de_pages)` : constructeur pour initialiser les attributs de Livre.
- `afficher_details(self)` : méthode pour afficher les détails spécifiques du livre (ISBN et nombre de pages).
- `__str__(self)` : méthode de représentation pour afficher les informations spécifiques du livre.

3. Classe Revue (hérite de Document)

La classe Revue représente une revue, un type de document périodique.

Attributs :

- numero (int) : le numéro de la revue.
- frequence (str) : la fréquence de publication (mensuelle, hebdomadaire, etc.).

Méthodes :

- **__init__**(self, id, titre, auteur, annee_publication, numero, frequence) : constructeur pour initialiser les attributs de Revue.
- **afficher_details**(self) : méthode pour afficher les détails spécifiques de la revue (numéro et fréquence).
- **__str__**(self) : méthode de représentation pour afficher les informations spécifiques de la revue.

4. Classe CD (hérite de Document)

La classe CD représente un CD, un type de document multimédia.

Attributs :

- genre_musical (str) : le genre musical du CD.
- duree (float) : la durée totale en minutes du CD.

Méthodes :

- **__init__**(self, id, titre, auteur, annee_publication, genre_musical, duree) : constructeur pour initialiser les attributs de CD.
- **afficher_details**(self) : méthode pour afficher les détails spécifiques du CD (genre et durée).
- **__str__**(self) : méthode de représentation pour afficher les informations spécifiques du CD.

5. Classe Bibliotheque

La classe Bibliotheque contient une liste de documents et offre des méthodes pour ajouter, supprimer, et mettre à jour des documents dans cette liste.

Attributs :

- documents (list) : une liste de documents (objets Document).

Méthodes :

- **__init__**(self) : constructeur pour initialiser la liste de documents vide.
- **ajouter_document**(self, document) : méthode pour ajouter un document à la bibliothèque.
- **supprimer_document**(self, id) : méthode pour supprimer un document par son identifiant.
- **mettre_a_jour_document**(self, id, titre=None, auteur=None, annee_publication=None) : méthode pour mettre à jour les détails d'un document spécifique (par identifiant).
- **afficher_documents**(self) : méthode pour afficher la liste de tous les documents.
- **filtrer_documents**(self, critere, valeur) : méthode pour filtrer les documents en fonction du critère (année ou auteur).
- **Serialiser** dans un fichier json et csv