

Objectif

Réaliser un mini projet en Scala permettant de gérer les documents d'une bibliothèque (livres, magazines, bandes dessinées), avec possibilité d'enregistrer des utilisateurs et de gérer les emprunts/retours.

Compétences visées

- Programmation orientée objet en Scala : classes, héritage, abstraction, traits
- Encapsulation et gestion d'état
- Polymorphisme
- Manipulation de collections

Spécifications

1. Types de documents

- a. Créer un **trait** *Borrowable* avec les méthodes :

```
def borrow(): Boolean
def returnItem(): Boolean
```

- b. Créer une **classe abstraite** *Document* qui implémente *Borrowable* et contient :

- Attributs communs (`title : String`, `author : String`, `year : integer`, `isBorrowed : Boolean`)
 1. Nous partons du principe qu'aucun document n'a été emprunté donc l'attribut `isBorrowed` a la valeur `false`.
- Une méthode abstraite `description(): String`, qui retourne une description pour voir l'état d'emprunt du document.
- Une implémentation par défaut de :
 1. `borrow()` : si le document est disponible (`isBorrowed == false`) return `true`, sinon `false`.
 2. `returnItem()` : si le document est déjà emprunté (`isBorrowed == true`) return `true`, sinon `false`.

c. Créer trois classes héritant de *Document* :

- *Book* avec un attribut **genre**
- *Magazine* avec un attribut **EditionNumber**
- *Comic* avec un attribut **seriesVolume**

Note: réimplémenter la méthode *description* pour afficher les informations sur les documents.

2. Gestion des utilisateurs

a. Créer une classe *User* qui contient :

- *name* : *String* Le nom de l'utilisateur
- *borrowedDocs* : *List[Document]* Une liste de documents empruntés
- Méthodes pour emprunter et retourner des documents
 1. *borrowDocument* : *Boolean* -> pour emprunter un document (utiliser la méthode *borrow* du document).
 2. *returnDocument* : *Boolean* -> pour retourner un document (utiliser la méthode *returnItem* du document).
 3. *listBorrowedDocuments* : *Boolean* -> afficher la liste des documents empruntés.

3. Gestion de la bibliothèque

a. Créer une classe *Library* qui contient :

- *documents* : *List[Document]* Une liste de documents.
- *users* : *List[User]* Une liste d'utilisateurs.
- Méthodes :
 1. *addDocument* : *Unit* -> ajouter un document
 2. *addUser* : *Unit* -> enregistrer un utilisateur.
 3. *listAvailableDocuments* : *Unit* -> afficher les documents disponibles.
 4. Des getters.

4. Programme principal

a. Dans un objet *Main*, instancier la bibliothèque, ajouter des documents et des utilisateurs, puis simuler quelques emprunts/retours.