

# Séries d'exercices POO

## Exercice 1

**Objectif** : modéliser une petite bibliothèque dans laquelle les utilisateurs peuvent emprunter, rendre et rechercher des livres.

Classes :

**Livre** : représente les livres individuels de la bibliothèque.

Attributs : titre, auteur, isbn, nbrExemplaire.

Méthodes : afficher\_details().

**Membre** : représente un membre de la bibliothèque qui peut emprunter des livres.

Attributs : nom, member\_id, livres\_empruntes (liste).

Méthodes : emprunter\_livre(livre), afficher\_liste\_emprunts().

**Bibliothèque** : gère la collection de livres et de membres.

Attributs : livres (liste d'objets Livre), membres (liste d'objets Membre).

Méthodes :

ajouter\_livre(livre) : ajoute un nouveau livre à la bibliothèque.

supprimer\_livre(livre) : supprime un livre de la bibliothèque.

enregistrer\_membre(membre) : enregistre un nouveau membre de la bibliothèque.

chercher\_livre\_par\_titre(titre) : recherche un livre par titre.

emprunter\_livre(membre, livre) : prêter un livre d'un membre.

retourner\_livre (membre, livre) : retourner un livre d'un membre.

## Exercice 2

**Objectif** : simuler un système bancaire avec différents types de comptes et autoriser les transactions entre comptes.

Classes :

**Compte** (classe de base) : classe de base pour tous les comptes bancaires.

Attributs : numero\_compte, solde.

Méthodes : deposer(montant), retrait(montant), afficher\_solde().

**CheckingAccount** (hérite de Compte) : compte courant avec frais de transaction.

Attributs supplémentaires : frais\_transaction.

Méthodes : remplace le retrait pour déduire les frais de transaction pour chaque retrait (2%).

**SavingsAccount** (hérite de Compte) : compte d'épargne avec intérêts.

Attributs supplémentaires : taux\_interet.

Méthodes : appliquer\_interet() : augmente le solde en fonction du taux d'intérêt.

**Client** : représente un client bancaire avec plusieurs comptes.

Attributs : client\_id, nom, comptes (liste des types de comptes).

Méthodes : ouvrir\_compte(type\_compte), fermer\_compte(type\_compte), afficher\_comptes().

**Banque** : gère tous les clients et toutes les transactions.

Attributs : clients (liste des objets Client).

Méthodes :

ajouter\_client(client), supprimer\_client(client).

transférer\_fonds(compte\_emetteur, compte\_destination, amount) : transfère des fonds entre les comptes.