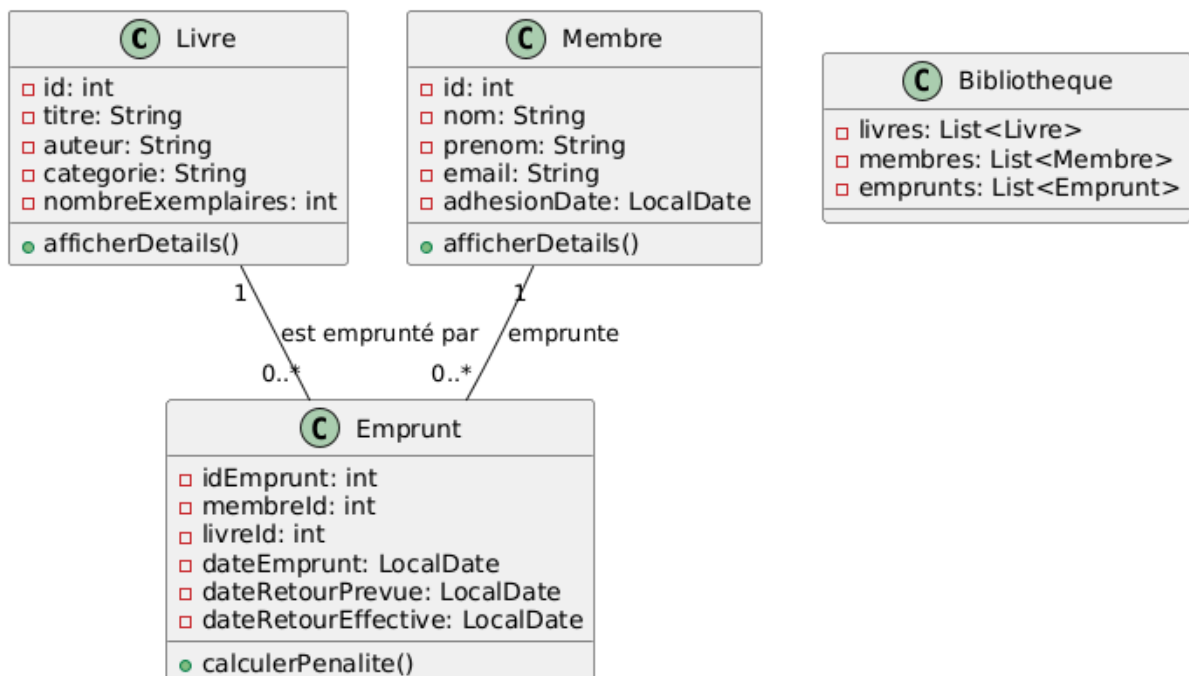


# DIADRAMME DE CLASSE DU SYSTEME DE GESTION DE BIBLIOTHEQUE



## Classes

### 1. Livre

- Description : Représente un livre dans la bibliothèque.
- Attributs :
  - `id: int` : Identifiant unique du livre.
  - `titre: String` : Titre du livre.
  - `auteur: String` : Auteur du livre.

- categorie : String : Catégorie du livre.
- nombreExemplaires : int : Nombre d'exemplaires du livre disponibles.
- Méthodes :
  - afficherDetails() : Affiche les détails du livre (titre, auteur, catégorie).

## 2. Membre

- Description : Représente un membre de la bibliothèque.
- Attributs :
  - id : int : Identifiant unique du membre.
  - nom : String : Nom du membre.
  - prenom : String : Prénom du membre.
  - email : String : Adresse e-mail du membre.
  - adhesionDate : LocalDate : Date d'adhésion du membre.
- Méthodes :
  - afficherDetails() : Affiche les détails du membre (nom, prénom, email).

## 3. Emprunt

- Description : Représente un emprunt de livre par un membre.
- Attributs :
  - idEmprunt : int : Identifiant unique de l'emprunt.
  - membreId : int : Identifiant du membre ayant emprunté le livre.
  - livreId : int : Identifiant du livre emprunté.
  - dateEmprunt : LocalDate : Date d'emprunt du livre.
  - dateRetourPrevue : LocalDate : Date de retour prévue du livre.
  - dateRetourEffective : LocalDate : Date de retour effective du livre.
- Méthodes :
  - calculerPenalite() : Calcule la pénalité en cas de retard de retour du livre.

## 4. Bibliotheque

- Description : Gère les opérations liées à la bibliothèque (livres, membres, emprunts).
- Attributs :
  - livres : List<Livre> : Liste des livres de la bibliothèque.
  - membres : List<Membre> : Liste des membres de la bibliothèque.
  - emprunts : List<Emprunt> : Liste des emprunts effectués.

- Méthodes :
  - ajouterLivre(Livre livre) : Ajoute un nouveau livre à la bibliothèque.
  - rechercherLivreParTitre(String titre) : Recherche un livre par son titre.
  - afficherLivres() : Affiche la liste de tous les livres de la bibliothèque.
  - inscrireMembre(Membre membre) : Inscrit un nouveau membre à la bibliothèque.
  - rechercherMembreParNom(String nom) : Recherche un membre par son nom.
  - enregistrerEmprunt(Emprunt emprunt) : Enregistre un nouvel emprunt.
  - gererRetourLivre(int idEmprunt, LocalDate dateRetour) : Gère le retour d'un livre emprunté.

## Relations

- Association :
  - Livre est associé à Emprunt avec une relation "1-N" (un livre peut être emprunté par plusieurs membres).
  - Membre est associé à Emprunt avec une relation "1-N" (un membre peut avoir plusieurs emprunts).
- Composition :
  - Bibliotheque contient des listes de Livre, Membre et Emprunt, indiquant une relation de composition.

## Conclusion

Ce diagramme de classes décrit la structure de l'application de gestion de bibliothèque, avec les principales entités (Livre, Membre, Emprunt) et leurs interactions. Il fournit une vue d'ensemble des fonctionnalités de l'application et peut servir de référence pour les développeurs travaillant sur ce projet.