

On souhaite gérer l'emprunt des livres dans une bibliothèque.

Soit la classe **abstraite** **Personne** qui comporte les attributs suivants :

- Le nom
- Le prénom

1. Définir la classe **Personne**.
2. Ajouter un constructeur par défaut et un constructeur paramétré qui initialise tous les attributs.
3. Ajouter les accesseurs et les mutateurs de chaque attribut :
4. Ajouter une méthode *toString()* qui retourne l'état d'un objet **Personne**.
5. Ajouter une méthode *equals()* qui teste l'égalité de deux personnes (deux personnes sont égales s'elles ont le même nom et le même prénom)

Soit la classe **Adherent** qui hérite de la classe **Personne** et qui a comme attributs :

- Le code de l'adhérent (doit être attribué automatiquement de manière incrémentale 1,2,...)
- La date d'adhésion

1. Définir la classe **Adherent**.
2. Ajouter un constructeur qui initialise tous les attributs de la classe **Adherent**.
3. Ajouter un constructeur paramétré qui initialise le nom et le prénom. Ce constructeur doit faire appel au constructeur précédant (la date d'adhésion est la date système).
4. Ajouter la méthode *toString()* qui retourne l'état d'un adhérent (respectez le format : nom_prenom_code_jj mm aaaa).

Soit la classe **Auteur** qui hérite de la classe **Personne** et qui a comme attributs :

- Le code de l'auteur (doit être un entier codé sur 4 chiffres)

1. Définir la classe **Auteur**.
2. Ajouter un constructeur qui initialise tous les attributs de la classe **Auteur** (une exception nommée **CodeAutException** doit être levée si le code de l'auteur est incorrect).
3. Ajouter les accesseurs et les mutateurs de chaque attribut.
4. Ajouter une méthode *toString()* qui retourne l'état d'un objet **Auteur**.

Soit la classe **Livre** qui comporte les attributs suivants :

- Le code du livre
- Le titre du livre
- L'auteur du livre
- Le nombre total des exemplaires
- Le nombre des exemplaires disponibles

1. Définir la classe **Livre**.

2. Ajouter un constructeur qui initialise tous les attributs de la classe Livre.
3. Ajouter une méthode *livreDisponible()* qui vérifie si un livre est disponible pour l'emprunt ou non.
4. Ajouter une méthode *retourLivre()* qui enregistre le retour d'un livre.

Soit la classe **Emprunt** qui comporte les attributs suivants :

- Le code de l'emprunt (attribué automatiquement)
- Le livre emprunté
- L'emprunteur du livre
- La date d'emprunt
- La date de retour prévue
- la date de retour effective

1. Définir la classe Emprunt.
2. Ajouter un constructeur qui initialise tous les attributs de la classe Emprunt (la date de retour est nulle).
3. Ajouter une méthode *etatEmprunt()* qui retourne l'état d'un emprunt (en cours, non rendu, rendu).
4. Ajouter une méthode *retourEmprunt()* qui enregistre le retour du livre emprunté.

Soit la classe **Biliotheque** qui comporte les attributs suivants :

- Une liste des livres
- Une liste des adhérents
- Une liste des emprunts

Remarque : pour la déclaration des listes on se base sur la structure de données Tableau (taille maximale et métadonnée pour spécifier le nombre des éléments)

1. Définir la classe Biliotheque.
2. Ajouter une méthode *rechercherAdherent()* qui recherche un adhérent par son code. Une exception nommée *AdherentInexistantException* doit être levée si le code adhérent est incorrect.
3. Ajouter une méthode *rechercherLivre()* qui recherche un livre par son code. Une exception nommée *livreInexistantException* doit être levée si le code livre est incorrect.
4. Ajouter une méthode *ajouterLivre(...)* qui permet d'ajouter un livre à la bibliothèque. Si le livre est a été déjà ajouté à la bibliothèque, il faut mettre à jour le nombre des exemplaires.
5. Ajouter une méthode *ajouterAdherent(...)* qui permet d'ajouter un adhérent à la bibliothèque.
6. Ajouter une méthode *ajouterEmprunt(codeA , codeL)* qui permet d'enregistrer un emprunt. La date de d'emprunt est la date système, la durée de l'emprunt est de 3 jours. Une exception nommée *AjoutEmpruntException* doit être levée si le code livre ou code adhérent est incorrect en spécifiant l'origine du problème.
7. Ajouter une méthode *retourLivre(codeEmprunt)* qui enregistre le retour d'un livre.