

Examen TP de Programmation Orientée Objet : Groupe 1

Durée : 1h30

On vous demande de concevoir un système de gestion d'une bibliothèque en utilisant les principes de l'héritage et du polymorphisme en java. Vous devrez développer des classes représentant différents types de livres, des abonnées et des emprunts. Le système doit permettre aux utilisateurs d'emprunter et de retourner des livres. Chaque livre peut être de type différent (un roman, ou un conte).

1) Ecrire les classe Livre, Roman, Conte et l'interface EtatLivre en Java définies comme suit :

Classe livre : classe non instanciable (4pts)

Chaque livre est identifié par un titre (String), un auteur (String), un ISBN (String), une disponibilité (boolean) et nbemprunt pour stocker le nombre d'emprunt.

La visibilité de tous les attributs est protected.

Méthodes :

- Constructeur avec quatre paramètres : Pour initialiser le titre, l'auteur, l'ISBN et la disponibilité et nbemprunt à zéro.
- Ecrire un autre constructeur à trois paramètres permettant de créer un nouveau livre et définir le statut de disponibilité par défaut à true (livre disponible). Ce constructeur appelle le constructeur avec quatre paramètres.
- Emprunter() : Change le statut de disponible à false si le livre est disponible et incrémente le nombre d'emprunt de 1, sinon affiche un message d'erreur.
- Retourner() : Change le statut de disponible à true lorsque le livre est retourné.
- Méthode toString()...

Interface EtatLivre : (1pts)

```
public interface EtatLivre {  
  
    void AfficherEtatLivre();}
```

L'interface EtatLivre sera implémentée par la classe Livre. L'interface inclut une seule méthode pour afficher le nombre d'emprunts et l'état d'un livre en fonction du nombre d'emprunts afin d'informer l'utilisateur de l'état du livre. L'état du livre est évalué sur la base du nombre d'emprunts :

- **Neuf** : 0 à 5 emprunts.
- **Bien conservé** : 6 à 15 emprunts.
- **Usé** : 16 à 30 emprunts.
- **Très usé** : Plus de 30 emprunts.

Classe Roman :(4pts)

La classe Roman représente un livre de type roman. Elle inclut un attribut supplémentaire spécifique aux romans, le genre (par exemple, "historique", "policier", etc.).

Méthodes :

- Constructeur pour initialiser tous les attributs y compris le genre (un roman est par défaut disponible).
- Equals (Roman autreRoman) : Compare deux romans en fonction de leur genre. Si les genres sont identiques, la méthode retourne true, sinon false.
- toString() : Redéfinie pour retourner une chaîne décrivant les informations du roman de manière spécifique, incluant le genre.

Classe Conte: (4pts)

Un conte représente un type spécifique de livre. La classe inclut un attribut supplémentaire :

- PublicCible (String) : Indique le public cible : "enfants", "adolescents", "adultes".

Méthodes :

- Le constructeur doit initialiser tous les attributs de la classe conte, ainsi que l'attribut publicCible. Un conte est par défaut disponible.
- Equals (Conte autreConte) : Compare deux contes en fonction du public cible. Si les deux PublicCible sont identiques, la méthode retourne true, sinon false.
- toString() : Redéfinie pour retourner une chaîne décrivant les informations du conte de manière spécifique, incluant le public cible.

Classe Abonne: (4pts)

- La classe Abonne permet de définir les personnes qui empruntent des livres. La classe possède des attributs nom_prenom (String), adresse (String), contact (int). L'attribut nom_prenom, une fois initialisé, ne peut plus changer de valeur.
- La classe Abonne possède également un attribut TabEmprunt un tableau de taille 10 pouvant contenir des livres polymorphes qui sont empruntés par l'utilisateur.

Méthodes :

- Un constructeur avec trois paramètres permettant d'initialiser tous les attributs de la classe Abonne et créer le tableau TabEmprunt .
- EmprunterLivre(Livre livre) : Cette méthode permet à un utilisateur d'emprunter un livre. Si le livre est disponible, il sera ajouté à la liste des livres empruntés et son statut de disponibilité sera mis à jour (le livre devient non disponible). Si le livre est déjà emprunté, un message d'erreur doit être affiché.
- RetournerLivre(Livre livre) : Cette méthode permet à un utilisateur de retourner un livre qu'il a emprunté. Le livre sera supprimé de la liste des livres empruntés et son statut de disponibilité sera mis à jour (le livre devient disponible).

- toString() : Pour retourner une chaîne décrivant un Abonné, la chaîne contient les informations détaillées de l'abonné, telles que son nom et prénom, adresse, numéro de contact et la liste des livres qu'il a empruntés.

Classe Bibliothèque : (3pts)

La classe Bibliothèque contient la méthode main.

2) Créez les livres et les abonnés suivants :

Les livres :

- Roman 1 : "Les Misérables" de Victor Hugo, ISBN : 978-1234, genre "Historique".
- Roman2 : "1984" de George Orwell, ISBN : "978-0451", genre "Dystopie".
- Conte 1 : "Le Petit Prince" d'Antoine de Saint-Exupéry, ISBN : 978-1238, public cible "enfants".
- Conte 2 : "La Belle au Bois Dormant" de Charles Perrault, ISBN : 978-1122, public cible "enfants".

Les abonnés :

- Abonné 1 : "John Doe", adresse "123 rue Principale", contact 123456789.
- Abonné 2 : "Jane Smith", adresse "456 rue Secondaire", contact 987654321.

3) Créez les emprunts effectués par les différents abonnés.

Emprunt de livres par les abonnés :

- Abonné 1 emprunte Roman 1 ("Les Misérables").
- Abonné 2 emprunte Conte 2 ("La Belle au Bois Dormant").
- Abonné 1 tente d'emprunter Conte 1 ("Le Petit Prince").

Retour de livres :

- Abonné 1 retourne Roman 1 ("Les Misérables").
- Abonné 2 retourne Conte 2 ("La Belle au Bois Dormant").

4) Effectuer les comparaisons suivantes et afficher un message :

- Comparer Roman 1 ("Les Misérables") avec Roman 2 ("1984") pour vérifier si le genre est le même.
- Comparer Conte 1 ("Le Petit Prince") avec Conte 2 ("La Belle au Bois Dormant") pour vérifier si le public cible est le même.

5) Affichage des informations sur les abonnés et les livres empruntés :

- Affichez les informations détaillées des deux abonnés et des livres qu'ils ont empruntés.
- Affichez l'état de tous les livres après plusieurs emprunts. Par exemple : Roman 1 ("Les Misérables") : Bien conservé, 2 emprunts.