

Module M2103 – Java

TP 2 : Premières Classes

Partie I

La librairie de l'université permet aux étudiants et aux enseignants d'emprunter des documents. Chaque document est identifié par son *code d'archivage* et son *numéro de copie*. Les autres informations à propos d'un document sont : son *titre*, son *auteur principal*, son *année de publication*. Les documents de la librairie sont présentés sur les étagères lorsqu'ils ne sont pas empruntés. Lorsqu'une personne retourne un document, celui-ci est placé temporairement sur la pile des retours avant d'être replacé sur son étagère.

Quand un document est emprunté, il est possible de faire une réservation sur celui-ci, ce qui signifie qu'il sera placé dans une section spéciale 'réservations' par l'agent de la librairie lorsqu'il est rendu au lieu d'être placé sur la pile des retours. Une seule personne peut faire une réservation pour un document, ce qui est possible seulement si le document est emprunté. La personne qui fait la réservation peut ensuite emprunter le livre lorsque celui-ci devient disponible ou annuler sa réservation à tout moment.

- **Tâche 1** – Créer la classe **DocLibrairie**, représentant un document (cette classe sera enrichie tout au long du TP)
- **Tâche 2** – Ecrire un constructeur de manière à pouvoir donner des valeurs pertinentes aux attributs de l'objet à l'instanciation.
- **Tâche 3** – Ecrire des accesseurs de manière à pouvoir obtenir de l'information à propos des attributs de l'objet.
- **Tâche 4** – Ecrire des mutateurs pour pouvoir par exemple corriger le nom de l'auteur, le code d'archivage...
- **Tâche 5** – Pour pouvoir utiliser la classe **DocLibrairie**, il faudra avoir une classe de test **TestLibrairie**. Oninstanciera quelques documents, comme par exemple ci-dessous. On affichera également de l'information à propos des documents en utilisant les accesseurs. Aussi, les mutateurs pourront être testés.

| Code d'archivage | Titre | Auteur | Année |
|------------------|-----------------------|------------|-------|
| 004. 178 K20PM | Introduction à Java | J. Leblanc | 2015 |
| 967. 4987 T248O | Structures de Données | M. Machin | 2013 |

- **Tâche 6** – Le système informatique de la librairie doit renseigner le nombre de livres empruntés, le nombre de livres sur la pile ainsi que le nombre de livres dans la section spéciale 'réservations'. Mettre à jour la classe **DocLibrairie** afin de prendre en compte ces trois indicateurs. Ecrire des méthodes permettant d'indiquer ces derniers.
- **Tâche 7** – Les documents peuvent être à leur emplacement dans la librairie, empruntés, sur la pile des retours ou dans la section spéciale 'réservations' en attente d'emprunt. Si le document est emprunté, alors il peut être réservé ou retourné à la librairie. S'il est retourné, il est d'abord placé sur la pile des retours puis remis à son emplacement dans la librairie au bout d'un certain temps (un libraire replace les documents à leur emplacement dans la librairie si une réservation n'a pas été réalisée. Si les documents ont été réservés, ils sont placés dans la section spéciale 'réservations'). Enfin, une réservation peut être annulée. Ecrire un ensemble de mutateurs afin de prendre en compte ces informations concernant un objet **DocLibrairie**.
- **Tâche 8** – Ecrire un accesseur qui détermine le statut d'un objet **DocLibrairie**. Aussi, vous pouvez implémenter des accesseurs pour déterminer si un objet se trouve dans un état particulier, par exemple le message **estEmprunte** déterminerait si un document particulier est emprunté ou non.
- **Tâche 9** – Enrichir la classe test pour tester les méthodes implémentées dans le cadre des tâches 7 et 8.

Enregistrer les classes ainsi mises en œuvre et les sauvegarder dans un dossier.

Partie II

La librairie de l'université doit également répertorier ses membres. L'information sauvegardée à propos des membres est : le nom du membre, son numéro de téléphone, son adresse ainsi que le numéro d'abonné du membre.

- **Tâche 1** – Créer la classe **MembreLibrairie**, représentant un membre. A cette étape, implémenter les attributs, constructeur (de manière à ce que toute l'information à propos du membre soit passée en paramètre) ainsi que les accesseurs appropriés.
- **Tâche 2** – Modifier les mutateurs implémentés dans le cadre de la partie I de manière à prendre en compte le fait qu'un membre de la librairie puisse emprunter un document ou faire une réservation. Vous aurez besoin de deux attributs permettant à un objet **DocLibrairie** de déterminer quel membre de la librairie emprunte le document ou fait une réservation sur celui-ci pendant qu'il est emprunté.
- **Tâche 3** – Ecrire des accesseurs de manière à pouvoir déterminer quel membre a emprunté ou a réservé un document (penser au type du résultat retourné). Que dire du cas où le document n'est ni emprunté, ni réservé?

Enregistrer les classes ainsi mises en œuvre et les sauvegarder dans un dossier.

Partie III

Jusqu'à maintenant, il n'a pas été possible pour l'utilisateur de votre programme de gestion de librairie d'interagir avec le système. Vous modifierez donc votre classe test de manière à ce qu'elle propose un menu interactif permettant à l'utilisateur du programme de :

- Afficher des informations sur un document : titre, code d'archivage...
- Connaître l'état d'un document (*sur les étagères, emprunté...*)
- Changer l'état d'un document (par exemple de l'état *sur les étagères* à l'état *emprunté...*)
- Faire une réservation ou l'annuler.
- Afficher le nombre de documents empruntés, sur la pile des retours ou dans la section spéciale 'réservations'.
- N'importe quelle autre opération que vous désirez inclure...

Vous pourrez considérer que les manipulations sont effectuées sur des objets déjà créés dans la classe test. Pour les plus téméraires, vous pourrez donner la possibilité à l'utilisateur de spécifier quels objets **DocLibrairie** ou **MembreLibrairie** seront utilisés pour une manipulation particulière (en présentant par exemple un menu de documents et/ou un menu de membres à utiliser).