



L'Utilisateur en retard

TP Objet 03 – bis

Projet Bookstore

Thèmes : une association, une méthode d'instance, diagramme de classe UML

Conseil : Pour gérer le temps, vous utilisez soit l'API Date-Time de Java8 (package java.time, java.time.format), soit les classes historiques Java7 des package java.util, java.text, ...

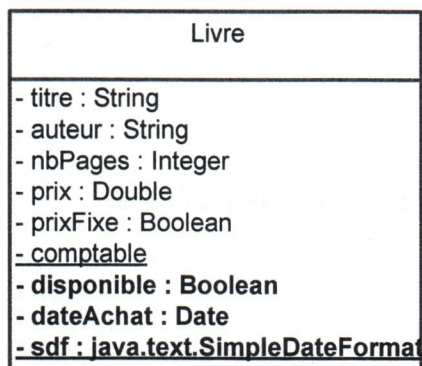
1 Compléter la classe Livre

Reprenez votre un projet Bookstore et, si nécessaire, ajouter au Livre les champs suivants :

- disponible qui est un booléen,
- dateAchat qui est une date-heure avec une précision 0 c.a.d. jusqu'à la seconde,
- dtf qui est un format pour afficher la date-heure (DateTimeFormatter)

Testez.

Illustration en Java7 :



2 La classe de l'Utilisateur

2.1 Travail à réaliser

Toujours dans le package domain, créez la classe de l'Utilisateur de la boutique Bookstore avec les attributs suivants :

- Le nom qui est une chaîne,
- Le prenom qui est une chaîne,
- L'idUtilisateur (identifiant de l'utilisateur) qui est un entier,
- « livre » qui est une référence vers le Livre emprunté (c'est une association),
- dateEmprunt : La date du dernier emprunt de Livre qui est une date sans l'heure,
- dureeMaxPret : durée maximum du prêt (en nombre de jour) et qui est un entier (champ static),
- dtf qui est un format pour la date (champ static)

Illustration en Java7 :

Utilisateur (from biblio.metier)
- nom - prenom - idUtilisateur - livre : Livre - dateEmprunt : Date - <u>dureeMaxPret : Integer = 15</u> - <u>sdf : SimpleDateFormat</u>

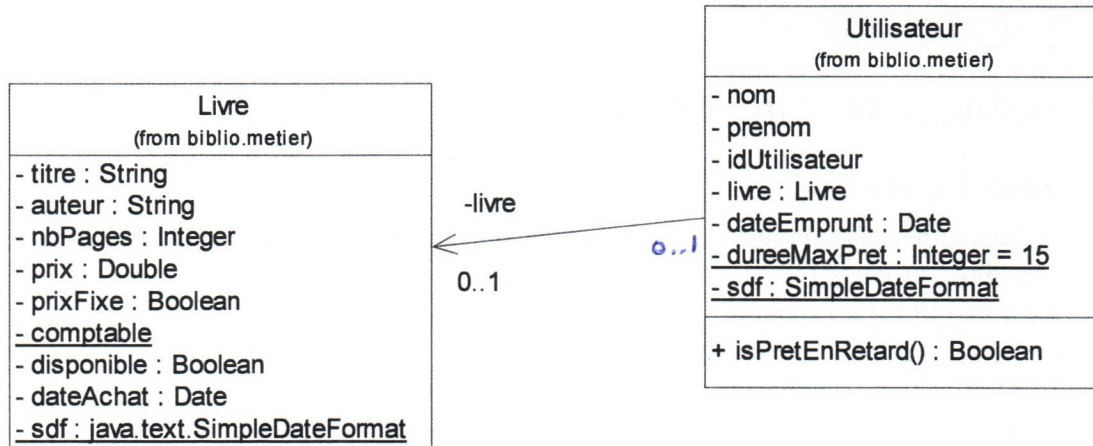
L'utilisateur ne peut emprunter qu'un seul livre. Il doit le rendre pour pouvoir en emprunter un nouveau. La variable de classe dureeMaxPret contient la durée maximum d'un emprunt (par ex : 15 jours) ; au-delà, l'utilisateur est considéré comme en retard.

Dans le main de Utilisateur, testez le scénario suivant :

1. Je remplit ma boutique Bookstore avec 2 livres asterix et tintin
2. Pour mon Bookstore, j'ai 2 utilisateurs: toto et zorro
3. toto emprunte le livre asterix
4. zorro emprunte le livre tintin
5. Les deux utilisateurs rendent leurs livres
6. toto emprunte le livre tintin

2.2 Annexe : Le diagramme de classe avec représentation de l'association

Illustration en Java7 :



3 La méthode d'instance isPretEnRetard()

C'est la responsabilité de l'Utilisateur de nous dire si son prêt courant est en retard ou pas. En vous inspirant d'une méthode isPretEnRetard que vous avez développée précédemment, et dans la classe Utilisateur, ajoutez une méthode d'instance isPretEnRetard() qui retourne un booléen indiquant si le prêt est en retard (ou pas).

Continuez le scénario de test, avec un emprunt en retard et un emprunt qui n'est pas en retard.