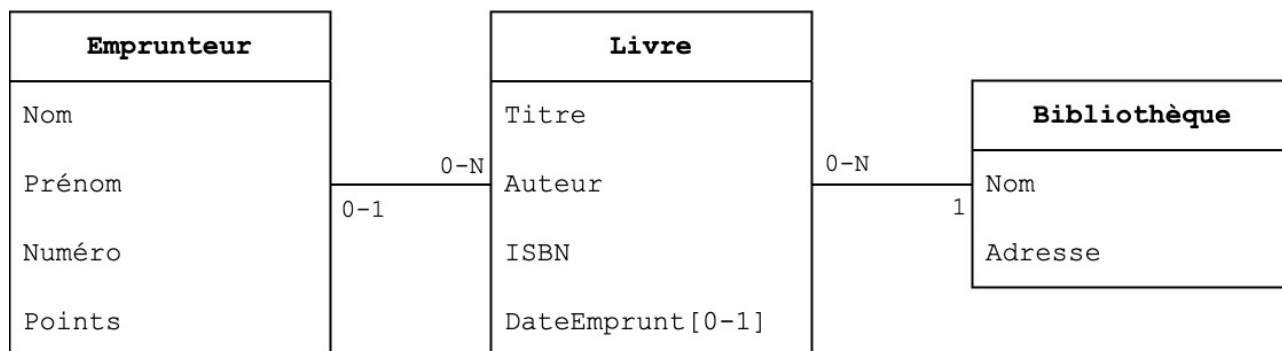


Programmation orientée objet (UAA14)

Exercice récapitulatif

Objets d'apprentissage :

- ✓ Modéliser une logique de programmation orientée objet.
- ✓ Déclarer une classe.
- ✓ Instancier une classe.
- ✓ Utiliser les méthodes de l'objet instancié.
- ✓ Traduire un algorithme dans un langage de programmation.
- ✓ Commenter des lignes de code.
- ✓ Tester le programme conçu.
- ✓ Caractériser les attributs dans une classe (encapsulation).
- ✓ Caractériser les méthodes dans une classe (encapsulation).
- ✓ Décrire la création d'un constructeur.
- ✓ Extraire d'un cahier des charges les informations nécessaires à la programmation.
- ✓ Programmer en recourant aux instructions et types de données nécessaires au développement d'une application.
- ✓ Corriger un programme défaillant.
- ✓ Développer une classe sur la base d'un cahier des charges en respectant le paradigme de la programmation orientée objet.
- ✓ Programmer en recourant aux classes nécessaires au développement d'une application orientée objet.
- ✓ Améliorer un programme pour répondre à un besoin défini.



Cet exercice vise à appliquer toutes les compétences acquises au cours des séries d'exercices précédentes sur la programmation orientée objet (POO) en PHP.

Imaginez que vous êtes développeur pour une application de gestion de bibliothèque. Votre mission est de concevoir un système de gestion des prêts de livres.

Un livre peut être emprunté par un seul emprunteur à la fois pour 25 centimes par jour. Lorsqu'un emprunteur emprunte un livre, celui-ci gagne 5 points de fidélité. Un emprunteur peut échanger ses points de fidélité contre une réduction sur ses emprunts : 10 points = 1 euros.

Classe Livre

Un livre est caractérisé par :

- un titre
- un auteur
- un ISBN
- la bibliothèque dans laquelle le livre se trouve
- un emprunteur, si le livre est emprunté
- la date de début de l'emprunt, si le livre est emprunté

Implémente un constructeur permettant de créer des livres. Lors de sa création, un livre n'est jamais emprunté.

Méthodes :

- `emprunter` : fais en sorte que le livre soit emprunté (à la date du jour) par un emprunteur reçu en argument
- `retourner` : fais en sorte que le livre ne soit plus emprunté par son emprunteur actuel et affiche le montant à payer (auquel on retire les points accumulés)
- `toString` : affiche toutes les informations sur le livre dont les informations de la bibliothèque dans laquelle il est stocké ainsi que les informations sur l'emprunteur (si le livre est emprunté)

Cas d'erreur (exceptions) :

- lors de la création d'un livre, une exception `ISBNIncorrectException` est lancée si l'ISBN fourni ne respecte pas le format standard
- si l'on tente d'emprunter un livre déjà emprunté, la méthode `emprunter` retourne une exception `LivreDejaEmprunteException`

Classe Emprunteur

Un emprunteur est caractérisé par :

- un nom
- un prénom
- un numéro
- des points de fidélité

Implémente un constructeur permettant de créer des emprunteurs. Lors de sa création, un emprunteur n'a pas de point de fidélité (0).

Méthodes :

- `ajouterPoints` : ajoute un certain nombre de points de fidélité (reçu en argument) à l'emprunteur ; si aucun nombre de point n'est spécifié, cette méthode ajoute 5 points par défaut
- `retirerPoints` : retire un certain nombre de points de fidélité (reçu en argument) à l'emprunteur
- `toString` : affiche toutes les informations sur l'emprunteur

Cas d'erreur (exceptions) :

- la méthode `retirerPoints` lance une `PointsInsuffisantsException` si l'emprunteur ne dispose pas assez de points

Classe Bibliothèque

Une bibliothèque est caractérisée par :

- un nom
- une adresse

Implémente un constructeur permettant de créer des bibliothèques.

Méthodes :

- `toString` : affiche toutes les informations sur la bibliothèque

Références

Les présents exercices ont été élaborés à l'aide des ressources suivantes :

- <https://www.pierre-giraud.com/php-mysql-apprendre-coder-cours/introduction-programmation-orientee-objet/>
- Cours de « Développement d'applications WEB », Hénallux (2019)