

TpOBJET_01

Projet Bookstore

Thèmes : Classes, attributs, constructeurs, accesseurs et mutateurs.
Méthode toString().

1 Création de la Classe Livre

Créer un projet Bookstore.

La classe Livre permet d'instancier des objets qui vont mimer le comportement d'un livre.

Voici le source de la classe Livre :

```
public class Livre {
    // Variables
    private String titre, auteur;
    private int nbPages

    // Constructeur
    public Livre(String unAuteur, String unTitre) {
        auteur = unAuteur;
        titre = unTitre;
    }

    // Accesseur
    public String getAuteur() {
        return auteur;
    }

    // Mutateur
    void setNbPages(int n) {
        nbPages = nb;
    }
}
```

Corrigez quelques petites erreurs dans le code java plus haut.

Mettez des commentaires javadoc sur les propriétés.

Ajoutez une méthode main () de Livre pour:

- Créez 2 livres,
- Affichez les auteurs de ces 2 livres.

2 Accesseurs et mutateurs

Modifiez la classe Livre :

- Ajoutez un accesseur pour la variable titre et la variable nbPages.
- Ajouter un mutateur pour les variables auteur et titre.
- Changez le mutateur de nbPages : il ne devra changer le nombre de pages que si on lui passe en paramètre un nombre positif, et ne rien faire sinon, en affichant un message d'erreur.

On remarquera l'intérêt des variables private : on peut contrôler leur modification dans les éventuelles méthodes qui permettent de les modifier.

Dans la méthode main () de Livre,

- forcez le nombre de pages de chacun des 2 livres,
- affichez ces nombres de pages,
- calculez le nombre de pages total de ces 2 livres et affichez-le.

3 La classe TestLivre et les packages

Ecrivez une classe `TestLivre` dans un fichier `TestLivre.java` qui comporte une méthode `main()` qui est un couper/coller de la méthode `main()` de la classe `Livre` précédente. Compilez et exécutez.

On adopte cette organisation pour la suite de l'exercice : 2 classes dans 2 fichiers source pour une application.

Placer la classe `Livre` dans le package `bookstore.metier`

Placer la classe `TestLivre` dans le package `bookstore.test`

4 Les méthodes `afficheToi` et `toString`

1. Dans la classe `Livre`, ajoutez une méthode `afficheToi()` qui affiche une description du livre (auteur, titre et nombre de pages). Utilisez `afficheToi()` dans la méthode `main()` de `TestLivre`.
2. Ajoutez l'instruction `System.out.println(livre)` où `livre` désigne un des livres que vous avez créés. Vous essaieriez de comprendre ce qui est affiché après avoir fait les 2 questions suivantes.
3. Ajoutez une méthode `toString()` qui renvoie une chaîne de caractères qui décrit le livre. Donnez à la méthode `toString()` le même profil que la méthode de même nom de la classe `java.lang.Object` (cherchez dans les API du JDK). Exécutez à nouveau la classe `TestLivre`. Voyez ce qui est affiché maintenant par l'instruction `System.out.println(livre)`. Miracle ! `println()` utilise automatiquement la méthode `toString()` de la classe de l'objet qu'il a à imprimer. Essayez de trouver une explication rationnelle en faisant la question suivante.
4. Il faut savoir chercher dans la documentation de l'API (javadoc). En partant de la classe `java.lang.System` et en cliquant sur les liens, retrouvez dans la documentation que `System.out.println(Objet)` affiche ce que retourne la méthode `toString` de la classe de `Objet`.
5. Modifiez la méthode `afficheToi()` pour utiliser `toString()`.

5 Le constructeur par défaut

Enlevez (provisoirement) le seul constructeur de la classe `Livre`. Sans ajouter de nouveau constructeur, peut-on quand même créer un nouveau livre dans la méthode `main` ? Si c'est possible, créez un livre écrit par Victor Hugo et faites afficher son titre et son auteur.

Remettez le constructeur que vous avez enlevé. Est-ce que le code de la méthode `main` de la question précédente fonctionne toujours (testez) ?

6 Surcharge du constructeur

Ici, les constructeurs utilisent:

- les accesseurs plutôt que l'accès directe,
- les autres constructeurs si possible.

Ajoutez un constructeur qui prend en paramètre l'auteur, le titre et le nombre de pages. Modifiez le constructeur qui prend en paramètre l'auteur et le titre du livre.

Ajoutez un constructeur sans paramètre.

Dans la méthode `main` de `TestLivre`, créez trois fois un livres de 300 pages avec un constructeur différent pour chacun.

Pour ceux qui auraient fini, ajoutez d'autres membres à la classe `Livre`.