

TP N°2

Objectifs :

1. Définir une classe avec attributs et méthodes
 2. Comprendre le principe de l'encapsulation et son intérêt
 3. Définir les méthodes spéciales : constructeurs, accesseurs et mutateurs
 4. Création et manipulation d'objets
 5. Manipulation d'une collection
 6. Relations entre classes
 7. Membres statiques
-

Dans une bibliothèque, on désire gérer les emprunts de livres. Un livre est caractérisé par son titre, auteur, nombre de pages et nombre d'exemplaires disponibles. Un livre a obligatoirement un titre et un nom d'auteur qui ne peuvent être modifiés. Le nombre d'exemplaires et le nombre de pages ne peuvent être négatifs. Le nombre d'exemplaires n'est modifié que par l'emprunt ou la restitution de livres.

1. Définir la classe Livre avec ses attributs.
2. Définir deux constructeurs de la classe Livre, le premier pour initialiser tous les attributs, le deuxième pour initialiser le titre et l'auteur avec un nombre d'exemplaires nul.
3. Définir les deux méthodes **emprunter()** et **rendre()**.
4. Définir une méthode **afficher()** qui affiche les informations relatives à un livre. Définir également la méthode **toString()** et montrer son utilisation dans le programme principal.
5. Définir les accesseurs (*Getters*) des attributs titre, auteur, nombre d'exemplaires.
6. Définir le *Setter* de l'attribut nombre de pages.
7. Définir une méthode pour comparer si deux livres sont similaires (même titre, même auteur).
8. Déclarer une nouvelle classe Main avec un programme principal qui teste les différentes méthodes conçues :
 - Créer les trois livres suivants :
 - Malek BENNABI, "Les conditions de la renaissance".
 - George ORWELL, "La ferme des animaux", 150 pages, 2 exemplaires.
 - Yahia SLIMANI, "Systèmes d'exploitation", 130 pages, 1 exemplaire.
 - Initialiser le nombre de pages du premier livre à 100.
 - Emprunter un exemplaire de chaque livre.
 - Afficher les livres.
9. Dans le programme principal définir un objet Bibliothèque comme une collection ArrayList de livres puis réaliser les traitements suivants :
 - Ajouter les 3 livres à la collection.
 - Afficher tous les livres de la collection.
 - Afficher les livres en rupture (nombre d'exemplaires = 0).

TP N°2 (Suite)

Ecrire en Java les classes Point et Cercle, puis les tester dans un programme principal :

I. Classe Point

1. Créer la classe Point ayant deux attributs privés x et y (de type double); représentant les coordonnées dans un plan d'un point donné.
2. Ajouter deux constructeurs, l'un initialisant les 2 attributs via des paramètres, le deuxième initialisant le point à l'origine (0.0 , 0.0).
3. Ajouter les méthodes getters et setters et la méthode **toString**.
4. Implémenter les méthodes **deplacer** et **distance** de deux manières différentes (méthodes d'objet puis méthodes de classe)
 - La méthode **deplacer** consiste à déplacer un point de ses coordonnées initiales (x, y) vers les coordonnées $(x + dx, y + dy)$: dx et dy sont donnés en paramètres.
 - La méthode **distance** permet de calculer la distance entre 2 points. Etant donné deux points de coordonnées respectives $(x1, y1)$ et $(x2, y2)$, la distance entre ces deux points est calculée avec la formule :

$$\sqrt{(x1 - x2)^2 + (y1 - y2)^2}$$

La distance ainsi calculée doit être retournée par la méthode.

II. Classe Cercle

1. Créer la classe Cercle ayant deux attributs privés, le centre de type Point et le rayon de type double.
2. Ajouter un constructeur permettant d'initialiser les attributs d'un cercle via des paramètres.
3. Ajouter deux méthodes permettant de retourner le périmètre et la surface d'un cercle.
 - Périmètre d'un cercle = $2\pi r$
 - Surface d'un cercle = πr^2
 - r étant le rayon (La valeur exacte de π est donnée par la constante **Math.PI**)
4. Ecrire une méthode booléenne permettant de tester si deux cercles sont concentriques (ayant le même centre).
5. Implémenter la méthode **toString**

III. Test des classes

1. Créer 2 points : O(0.0 , 0.0) et A(3.0 , 5.0).
 2. Afficher la distance entre les deux points O et A.
 3. Déplacer le point A vers les coordonnées (4.0 , 2.0).
 4. Afficher la nouvelle distance entre les deux points O et A.
- Pour les questions 2,3 et 4 utiliser la variante méthodes d'objet puis la variante méthodes de classe.
5. Créer trois cercles :

Objet	Centre	Rayon
C1	O	5.0
C2	A	3.0
C3	O	3.0

6. Afficher les 3 cercles avec leur surface et périmètre.
7. Vérifier que les cercles C1 et C3 sont concentriques. Essayer avec les cercles C1 et C2.