

TP 1 JAVA OO

Exercice 1

Soit la class Personne définit par les champs suivant :

- CIN : String
- Nom : String
- Prénom : String
- Email : String
- Age : Int

1. Écrire le code pour créer la class Personne
2. Écrire la fonction initialiser qui permet d'initialiser les valeur d'un objet personne
3. Écrire une fonction afficherPersonne qui permet d'afficher les informations d'une personne
4. Écrire une fonction main pour implémenter un exemple d'utilisation

Exercice 2 :

Réaliser une classe Point permettant de représenter un point sur un axe. Chaque Point sera caractérisé par un nom (de type char) et une abscisse (de type double).

On prévoira :

- Une fonction initialiser recevant en arguments le nom et l'abscisse d'un point,
- Une deuxième fonction initialiser recevant en argument un objet de type Point
- une méthode affiche imprimant le nom du point et son abscisse,
- une méthode translate effectuant une translation définie par la valeur de son

argument.

Écrire un petit programme utilisant cette classe pour créer un point, en afficher les caractéristiques, le déplacer et en afficher à nouveau les caractéristiques.

Exercice 3 :

Réaliser une classe Point permettant de représenter un point sur un axe. Chaque Point sera caractérisé par un nom (de type char) et une abscisse (de type double).

On prévoira :

- Un constructeur recevant en arguments le nom et l'abscisse d'un point,

- une méthode affiche imprimant (en fenêtre console) le nom du point et son abscisse,
- une méthode translate effectuant une translation définie par la valeur de son

argument.

Écrire un petit programme utilisant cette classe pour créer un point, en afficher les caractéristiques, le déplacer et en afficher à nouveau les caractéristiques.

Exercice 4

1. Écrire une classe Livre avec les attributs suivants :titre, auteur, nombre de pages et prix. Cette classe contiendra deux constructeurs :
 - Un constructeur qui prend en paramètre le titre, l’auteur et le nombre de pages.
 - Un deuxième constructeur qui a un paramètre de plus qui est le prix.
2. Des méthodes d’accès et d’altération (get et set) pour les différents attributs.
3. Une méthode toString() qui renvoie une chaîne de caractères qui décrit le livre. Si le prix d’un livre n’a pas été donné, la description du livre (toString()) devra indiquer «Prix pas encore donné ».

Un livre peut être gratuit (0 est une valeur possible pour un prix) ; la valeur -1 indiquera que le prix n’a pas encore été donné. On bloque complètement les prix : un prix ne peut être donné qu’une seule fois et ne peut être modifié ensuite (une tentative pour changer le prix ne fait qu’afficher un message d’erreur). Vous ajouterez une variable booléenne prixFixe qui indiquera que le prix ne peut plus être modifié.

4. Écrivez une méthode « prixFixe » qui renvoie si le prix a déjà été fixé.
5. Écrivez une méthode « compare » pour comparer 2 livres sur leur nombre de pages. Cette méthode prend un livre en paramètre, et elle renvoie 0 si les deux livres ont le même nombre de pages, 1 si l’instance courante (this) a plus de pages que le paramètre et -1 sinon.
6. Créez une classe **Etagere** pour représenter une étagère qui peut contenir un certain nombre de livres (fixe pour chaque étagère). Vous utiliserez pour cela un tableau.
7. Le constructeur prendra en paramètre le nombre de livres que pourra contenir l’étagère.
8. Vous ajouterez des méthodes pour :
 - a. Donner le nombre de livres que peut contenir l’étagère, et le nombre de livres qu’elle contient.

- b. Ajouter des livres (« void ajouter (Livre) »). Vous ajouterez les livres « à la fin » de l'étagère. Il devra être impossible d'ajouter des livres dans une étagère pleine.
- c. Récupérer un livre dont on donne la position sur l'étagère (le livre reste sur l'étagère, on récupère simplement les informations sur le livre). La méthode renverra une instance de livre. La position du premier livre d'une étagère devra être 1 (et pas 0, bien que le livre soit rangé dans la première position du tableau, qui est d'indice 0). La signature de la méthode sera « Livre getLivre(int) ».
- d. Chercher sur l'étagère un livre repéré par son titre et son auteur. La méthode renverra la position du livre dans l'étagère (ou 0 si le livre n'y est pas). Le profil de la méthode sera.
- e. Supprimer des livres.