

Série TP N° 01: notions de base POO

Exercice 01:

Écrire un programme qui calcule le PGCD (Plus Grand Diviseur Commun) de deux entiers a et b saisi au clavier. Utilisez l'algorithme d'Euclide qui consiste à soustraire successivement le plus petit au plus grand des deux entiers jusqu'à ce qu'ils soient égaux. Par exemple, pour a = 56 et b = 35 :

1. $56 - 35 = 21$
2. $35 - 21 = 14$
3. $21 - 14 = 7$
4. $14 - 7 = 7$ (C'est le PGCD)

Le programme contient deux méthodes :

1. La méthode *main()*
2. Une méthode *int pgcd(int a, int b)* qui retourne le PGCD de a et b.

Exercice 02:

1. Écrire un programme *TestSurface* qui construit un objet **Rectangle** ensuite calcule et affiche sa surface. Utiliser les méthodes *getWidth* et *getHeight*.
2. Écrire un autre programme *TestPerimetre* qui construit un objet **Rectangle** ensuite calcule et affiche son périmètre. Utiliser les méthodes *getWidth* et *getHeight*.

Exercice 03:

Implémentez une classe «livre» . un livre est connu par numéro ISBN, un titre, le nom de son auteur, nbrCopiesDisponibles , et un état qui indique si le livre est emprunté ou non;

1. Fournir les constructeurs :

Public Livre(int isbn,String Titre, String auteur, int nbrCopiesDisponibles, boolean etat)

Public Livre(String Titre,int nbrCopiesDisponibles)

2. Définir les getters des attributs titre, auteur, nbrCopiesDisponibles,et la méthode **afficher()** qui permet d'afficher les attributs du livre.
3. Définir les setters du titre et de l'auteur du livre.
4. Définir une méthode qui vérifie si deux livre sont du même auteur.

5. Définir les deux méthodes qui permettent d'emprunter et de rendre le livre.
6. Écrire un programme TestLivre qui permet d'instancier deux livres et tester toutes les méthodes.

Exercice 04:

On veut modéliser un vecteur 3d permettant de manipuler des vecteurs à trois composantes (de type double) et disposant de :

1. Un constructeur sans arguments permettant d'initialiser les coordonnées d'un vecteur aux valeurs: 3.0, 5.0, et 7.5.
2. Un constructeur à trois arguments.
3. Une méthode **afficher()**, permettant d'afficher les coordonnées d'un vecteur sous la forme: [**composante_1,composante_2,composante_3**].
4. Une méthode **compter()** qui permet de compter le nombre de vecteurs créés à partir de cette classe.
5. Une méthode qui calcule la norme d'un vecteur. Tel que **norm**= $\sqrt{x^2 + y^2 + z^2}$.
6. Une méthode (statique) fournissant la somme de deux vecteurs.
7. Une méthode (non statique) fournissant le produit scalaire de deux vecteurs. Tel que le produit scalaire est un nombre égal à la somme des produits de leurs composantes respectives.
8. Dans un programme TestVecteur, créer deux vecteurs vec1 et vec2, calculer et afficher leur somme et leur produit scalaire.