



Atelier 5

Objectif : Pratiquer La programmation POO en Python.

Partie 1 : généralités

1. Définir une classe **Vecteur2D** avec un constructeur fournissant les coordonnées par défaut d'un vecteur du plan (par exemple : $x = 0$ et $y = 0$).

Dans le programme principal, instanciez un **Vecteur2D** sans paramètre, un **Vecteur2D** avec ses deux paramètres, et affichez-les.

2. Enrichissez la classe Vecteur2D précédente en lui ajoutant une méthode d'affichage et une méthode de surcharge d'addition de deux vecteurs du plan.

Dans le programme principal, instanciez deux Vecteur2D, affichez-les et affichez leur somme.

3. Définir une classe Rectangle avec un constructeur donnant des valeurs (longueur et largeur) par défaut et un attribut nom = "rectangle", une méthode d'affichage et une méthode surface renvoyant la surface d'une instance.

Définir une classe Carre héritant de Rectangle et qui surcharge l'attribut d'instance : nom = "carré".

Dans le programme principal, instanciez un Rectangle et un Carre et affichez-les.

4. Définir une classe Point avec un constructeur fournissant les coordonnées par défaut d'un point du plan (par exemple : $x = 0.0$ et $y = 0.0$).

Définir une classe Segment dont le constructeur possède quatre paramètres : deux pour l'origine et deux pour l'extrémité. Ce constructeur définit deux attributs : orig et extrem, instances de la classe Point. De cette manière, vous concevez une classe composite : la classe Segment est composée de deux instances de la classe Point.

Ajouter une méthode d'affichage.

Enfin écrire un auto-test qui affiche une instance de Segment initialisée par les valeurs 1, 2, 3 et 4.

Partie 2 : Collections

Créer une classe étudiant avec les attributs « nom , prenom, age , cne , moyenne », puis créer une liste de type étudiant.

Essayer d'ordonner cette liste selon le age, moyenne.

Partie 3 : Diagramme de class

Traduire le digramme de classe suivant en code source python.

