

TP 3 Java  
Les interfaces et classes abstraites

**Exercice1 :**

L'objectif de cet exercice est de manipuler des formes géométriques sous forme de classes java. Toutes les formes géométriques possèdent des attributs et des comportements communs. Ceux-ci peuvent être définis dans une classe abstraite **Forme** qui sera héritée par les différentes classes concrètes représentant les différentes formes géométriques.

1. Définir la classe abstraite **Forme** contenant deux attributs réels représentant la position du centre. On définira aussi les getters et les setters ainsi qu'une méthode abstraite **afficher()** permettant d'afficher une forme géométriques.
2. Définir la classe **Rectangle** qui contiendra en plus de la position du centre la largeur et la hauteur.
3. Définir la classe **Carre** qui contiendra en plus de la position du centre la longueur d'un côté.
4. Définir la classe **Triangle** qui contiendra en plus de la position du centre la base et la hauteur. On considère qu'un triangle est toujours isocèle.
5. Définir la classe **Cercle** qui contiendra en plus de la position du centre le rayon.
6. Définir l'interface **Déformable** qui correspond aux Formes qu'on peut déformer horizontalement et verticalement de manière à obtenir une nouvelle Forme. Cette interface contiendra la méthode **Forme déformation(double coeffH, double coeffV)** où **coeffH** est le coefficient de déformation horizontale, et où **coeffV** est celui de déformation verticale.
7. Implémentez l'interface **Déformable** dans toutes les classes ou c'est possible.
8. Ajouter les méthodes suivantes dans la classe **Forme** et préciser si elles doivent être abstraites.
  - a. **double estDistantDe(Forme f)** qui calculera la distance entre le centre de la forme courante et le centre de **f**
  - b. **double surface()** qui calculera la surface de la Forme.

- c. Void `deplacement(int x, int y)` qui déplace une forme (et la modifie donc) dans la direction indiquée par x et y.

## **Exercice 2 : projet bibliothèque**

Reprenez le TP mediatheque et ajoutez des interfaces et des classes abstraites autant que nécessaire.