

## TD 3.1 : Creation de classes simples

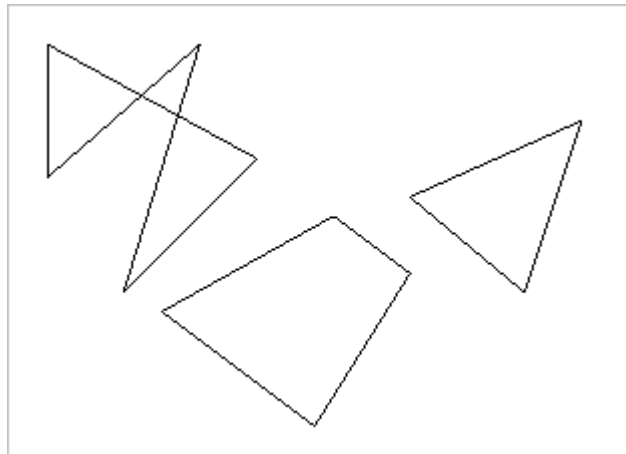
La première approche de création de classes simples est souvent liée à un contexte visualisable : la description d'une géométrie.

Nous nous servirons de cet exemple classique pour construire des premières classes.

### Identification des "objets" du système

On cherche à modéliser (activité constante de la programmation objet qui consiste à trouver les objets qui représentent un problème) la représentation de figures géométriques dans un dessin. Les collections d'objets seront toutes résolues par des tableaux statique pour l'exercice.

On dispose du dessin suivant :



#### Questions et actions (feuille de route)

Quels sont les objets qui pourraient suffire à décrire cette figure ?

Trouvez la "définition" de chacun des objets, c'est à dire, une phrase qui décrit ce qu'ils représentent.

Trouver les champs nécessaires à la description de chacun des objets.

### Définition et écriture des classes

Définir les enveloppes de classes correspondant à ces définitions.

Déterminer les besoins de construction de ces objets. Ecrire le ou les [constructeurs](#) intéressants pour chaque classe.

On rajoute une petite "vraie méthode", pour afficher le contenu de tous ces objets. Pour fabriquer cette possibilité d'affichage, il faut tenir compte que certains objets en contiennent d'autre. On commencera par les objets les plus simples.

Le principe général est d'écrire une méthode `toString()` (qui existe déjà par ailleurs, mais on ne s'en occupe pas) qui renvoie une expression "affichable" du contenu de l'objet).

### Mise en place d'une application et test

La séquence finale doit permettre de tester ce que nous avons fait. Il faut donc écrire une méthode `main()` statique et publique quelque part. La classe `Figure` peut être une candidate pour cette méthode `main()`. Dans `main()` on doit construire des objets des types précédents et les assembler correctement pour fabriquer une figure. Puis on imprime la figure.