

---

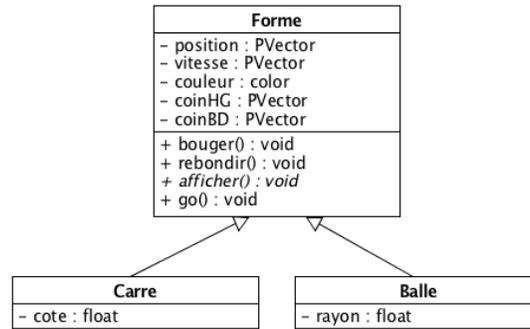
# POO - TD n°3

## Héritage

S3 / GEII / IUT Troyes - B. Jacquot / F. Morain-Nicolier

---

Ce TD est consacré à la création d'une hiérarchie de classe décrivant des formes (balles et carrés). Deux classes **Carre** et **Balle** hériteront d'une classe **Forme** comme spécifié dans le diagramme UML suivant :



- La méthode `afficher()` est en italique pour indiquer qu'elle est abstraite : elle doit être implémentée dans les sous-classes.
- Les méthodes `bouger()`, `rebondir()`, `go()` sont similaires au TD précédent, de même que les attributs `position`, `vitesse` et `couleur`.
- Les attributs `coinHG` et `coinBD` représentent les deux coins d'une boîte rectangulaire englobant la forme. La gestion des rebonds sur les bords sera basée sur ce rectangle.

Le programme principal sera par exemple le suivant :

```
Forme f1, f2;

void setup() {
    float r, a;
    PVector p, v;
    color c;

    c = color(100, 100, 0);
    p = new PVector(15, 20);
    v = new PVector(1, 1);
    r = 10;
    f1 = new Balle(p, v, c, r);

    c = color(0, 100, 0);
    p = new PVector(18, 20);
    v = new PVector(0.5, 1);
    a = 20;
    f2 = new Carre(p, v, c, a);
}

void draw() {
    background(255);
    f1.go();
    f2.go();
}
```

### Exercice 1

Écrire la classe **Forme** et les sous-classes **Carre** et **Balle** correspondant aux spécifications.

### Exercice 2

En toute rigueur, **Forme** doit être une classe abstraite puisqu'elle contient une méthode « vide » qui doit être implémentée par ses sous-classes (la méthode **afficher()**) :

- Dans la classe **Forme**, modifier la déclaration de **afficher()** en **abstract afficher()**, exécuter votre programme et observer le message d'erreur. Corriger l'erreur en modifiant la déclaration de la classe.
  - Modifier la classe **Balle** en supprimant l'implémentation de la méthode **afficher()** et observer le message d'erreur.
- ⇒ *La déclaration de classes et de méthodes abstraites permet de forcer leur implémentation dans les sous-classes.*

### Exercice 3

Enrichir votre hiérarchie de classes avec d'autres formes. Mettre à jour le diagramme UML pour qu'il corresponde.