

# Binding, initialisations

Leleux Laurent

2022 - 2023

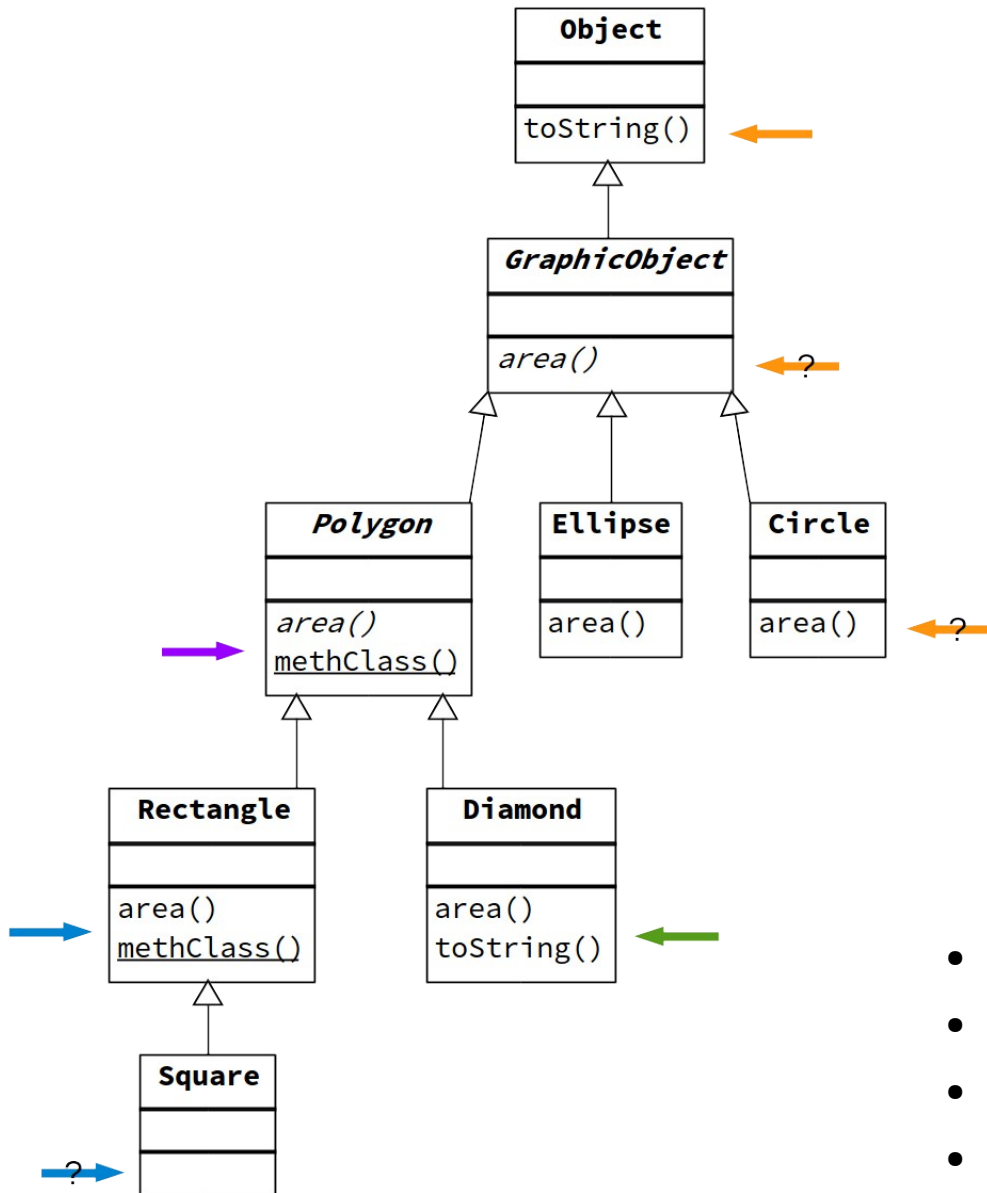
# Types

- Type d'une instance
  - Classe
  - Hiérarchie d'héritage
  - Interfaces implémentées
- Ensemble des membres visibles

# Static vs dynamic « Binding »

- Static binding
  - A la compilation
  - Méthodes de classe
  - Direct, en fonction du type
- Dynamic binding
  - A l'exécution
  - Méthodes d'instance
  - Recherche dans la hiérarchie d'héritage

# Example



- `circle.toString()`
- `diamond.toString()`
- `squareAsPolygon().area()`
- `squareAsPolygon().methClass()`<sup>4</sup>

# Class Initialization

- `<clinit>`
- Blocs statiques anonymes
- Pas de paramètres ni de type de retour
- Nombre illimité
- Compilés en un bloc unique

# Example

```
class Dog {  
  
    private static List<Dog> dogs = new ArrayList<Dog>();  
    private static int number;  
  
    static {  
        Dog.dogs.add(new Dog("Derek"));  
        Dog.dogs.add(new Dog("Gérard"));  
    }  
  
    static {  
        Dog.number = Dog.dogs.size();  
    }  
  
}
```

# Instance Initialization

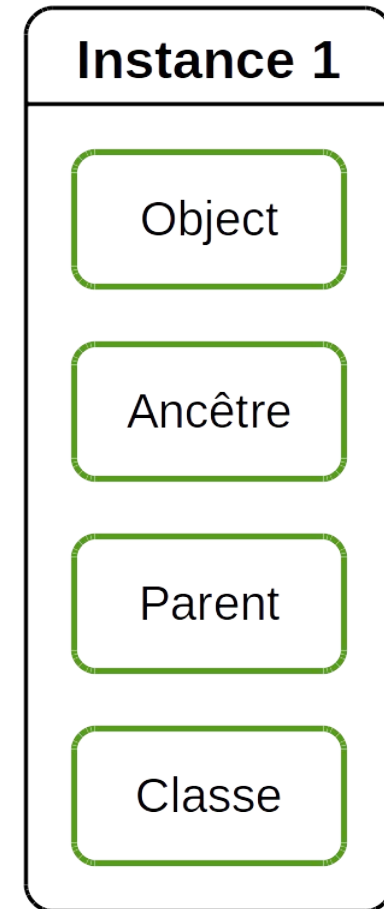
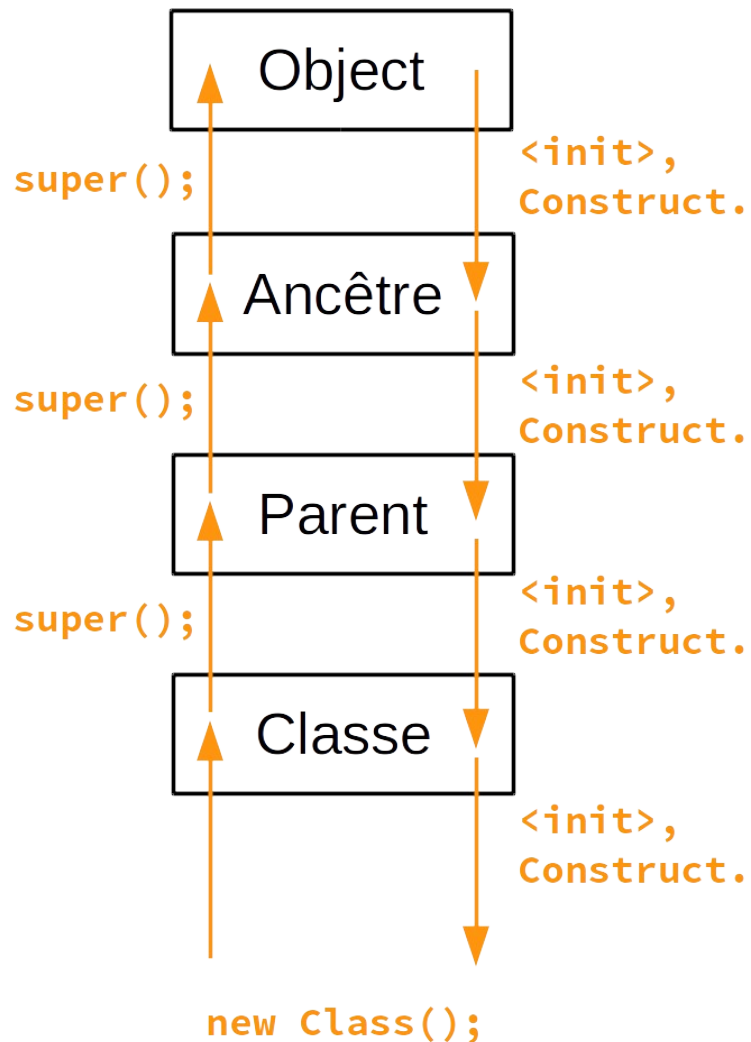
- Constructeur + <init>
- Blocs anonymes
- Pas de paramètres ni de type de retour
- Nombre illimité
- Compilés en un bloc unique

# Example

```
class Dog {  
  
    private String color;  
    private int age;  
    private List<String> vaccines = new ArrayList<String>();  
  
    {  
        this.vaccines.add("Rage");  
    }  
  
    public Dog(String color, int age) {  
        this.color = color;  
        this.age = age;  
    }  
  
    public Dog(String color) {  
        this.color = color;  
    }  
  
}
```



# Enchaînement des constructeurs



# New « Class Data » structure

