

# Classes, instances

Leleux Laurent

2022 - 2023

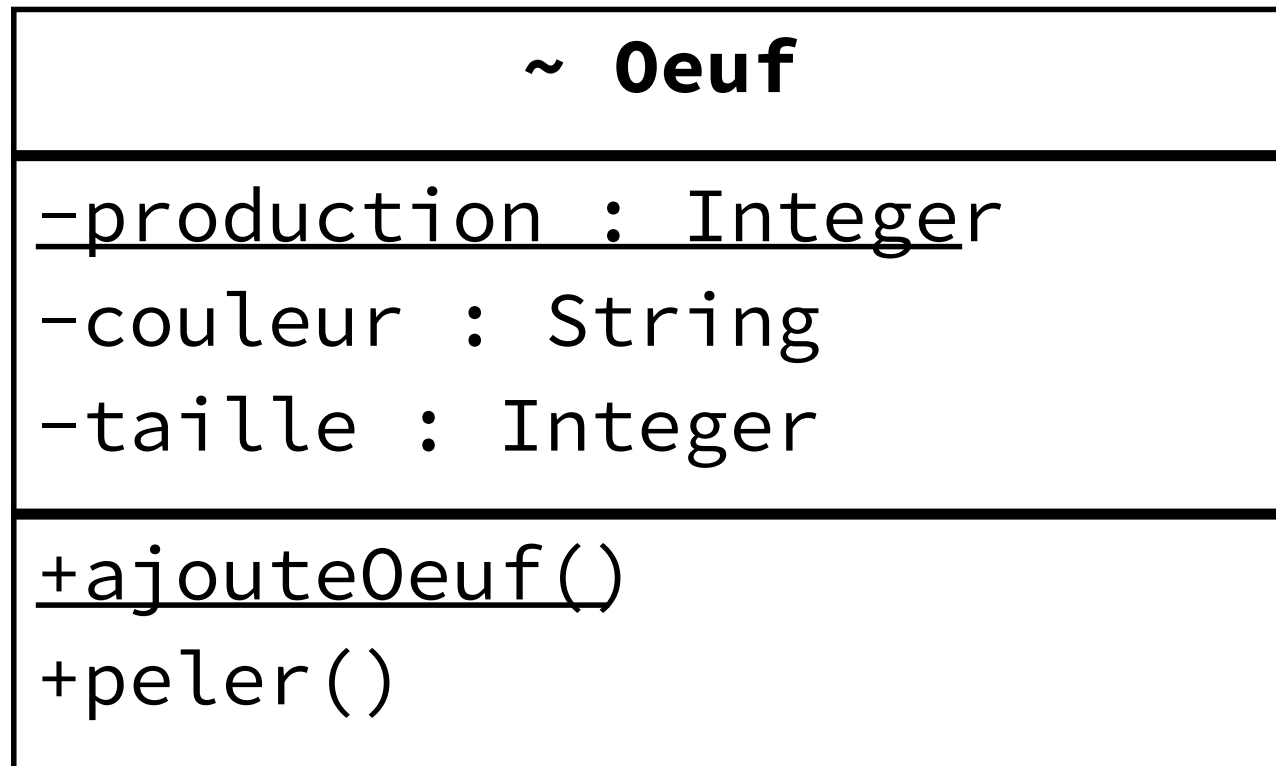
# Classe - Instance

- Classe
  - Usine à instances
- Instance
  - Sac de données

# Example

```
class Oeuf {  
  
    private static int production = 0;  
    private string couleur;  
    private int taille;  
  
    public Oeuf(string couleur, int taille) {  
        this.couleur = couleur;  
        this.taille = taille;  
    }  
  
    public static int ajouteOeuf() {  
        return ++Oeuf.production;  
    }  
  
    public void peler() {  
        // Something...  
    }  
  
}
```

# En UML

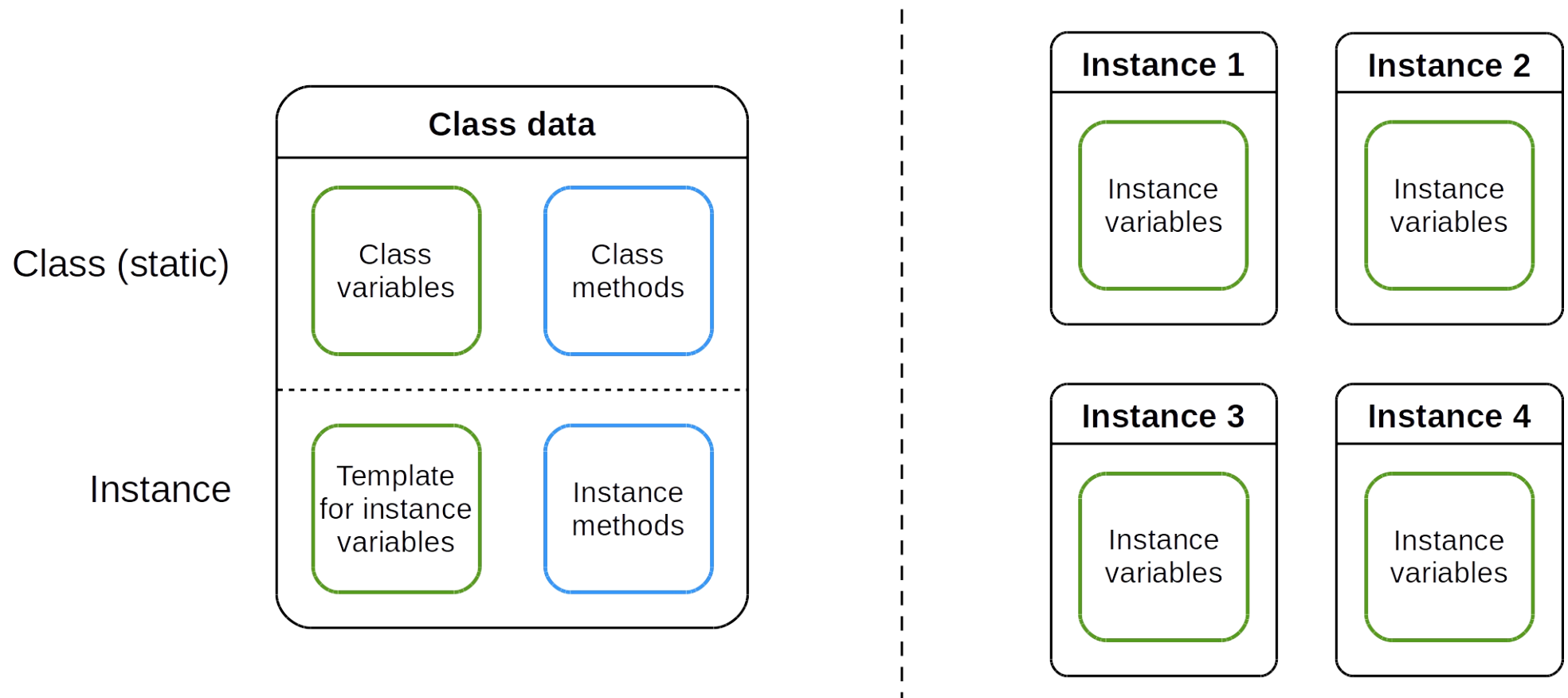


# Séparation

- Méthodes de classe
- Méthodes d'instance
- Attributs de classe
- Attributs d'instance

Template != valeurs !

# Classe - Instance



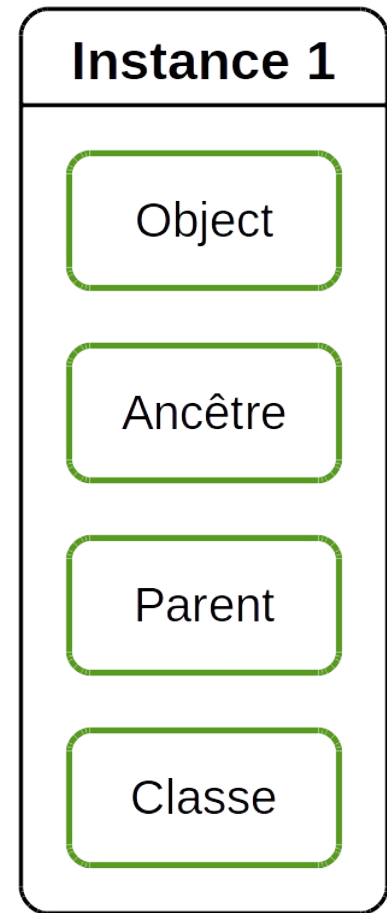
# State - Behaviour

- State
  - L'ensemble des valeurs des attributs de l'instance
- Behaviour
  - L'ensemble des méthodes qui s'appliquent sur l'instance
- Stateless

# Héritage ?

« Une instance,  
c'est comme un oignon »

« Le sac de données  
à plusieurs couches »





# Constructeurs ?

- +- méthode de classe
- +- méthode d'instance
- Signature + corps
- Pas de type de retour
- Visibilité
- Pas d'héritage

# Nouvelle instance ?

```
Oeuf oeuf = new Oeuf("brun", 7);
```

- Nouveau sac
- Référence disponible
- Constructeur
- Retour de la référence

# Héritage et constructeurs

- `super()`
  - Implicite ?
  - Explicite ?
  - Obligatoire...
- `this()`

# Enchaînement des constructeurs

