

**Programmation, bases de données et  
serveurs-AEC(LEA.D4)  
Technique de l'informatique DEC  
Intensif(420.B0)**

**Introduction à la programmation  
Orientée Objet  
420-W20-SF  
Pondération: 3-3-3  
Hiver 2020**

# Introduction aux principes de la POO

# Les constructeurs

# Objectifs

- À la fin de cette leçon, l'étudiant sera en mesure de:
  - Comprendre le rôle du constructeur
  - Connaître les propriétés du constructeur
  - Créer différents types de constructeurs
  - Invoquer le constructeur lors de la création des objets

# Rappels

- Qu'est-ce qu'un accesseur?
- Quels sont les deux types d'accesseurs?
- Qu'est-ce qu'une Association?
- Qu'est-ce qu'une composition?
- Qu'est-ce qu'une agrégation?
- Dire ce qu'est un diagramme de séquence?

# Le Constructeur

- Le **constructeur** d'une classe est une pseudo-méthode, qui n'a pas de type de retour, porte le nom de la classe et est appelée à l'aide du mot-clé **new**.
- La tâche du constructeur est d'initialiser la mémoire allouée à l'objet, où ses champs seront stockés.

# Propriétés du constructeur

- Il a le même nom que la classe
- Il n'a pas de type, ne retourne aucune valeur, ni même le type void
- Une classe peut avoir plus d'un constructeur
- Les constructeurs doivent différer par leur signature:
  - Nombre de paramètres formels
  - Types des paramètres formels

# Exemple

```
public class ExempleBonConstructeur
{
    // Bon constructeur
    public ExempleBonConstructeur()
    {
    }
    // Mauvaise déclaration du Constructeur ou de la
méthode.
    private string ExempleBonConstructeur()
    {
        return "Mauvais Constructeur de la classe!";
    }
}
```



# Déclaration

- On distingue deux types de constructeur:
  - Constructeur par défaut (sans paramètres)
  - Constructeur d'initialisation (avec paramètres)
- Formellement, la déclaration est la suivante:
  - [`<modificateur>`]`<nom_classe>`(`[<liste_paramètres>`])
  - Ex: `public Chien()`  
`{`  
`}`

# Déclaration

```
class inventaire
{
    public inventaire()
    { }
    public inventaire(string nom)
    {

    }
    public inventaire(string nom, int nombreitem, double prix)
    {

    }
    public inventaire(string nom, int nombreitem, double prix, int nombreenstock)
    {

    }
    //Ajoutez les méthodes additionnelles ici
    private string nom;
    private int nombreitem;
    private double prix;
    private int unitesEnStock;
}
```

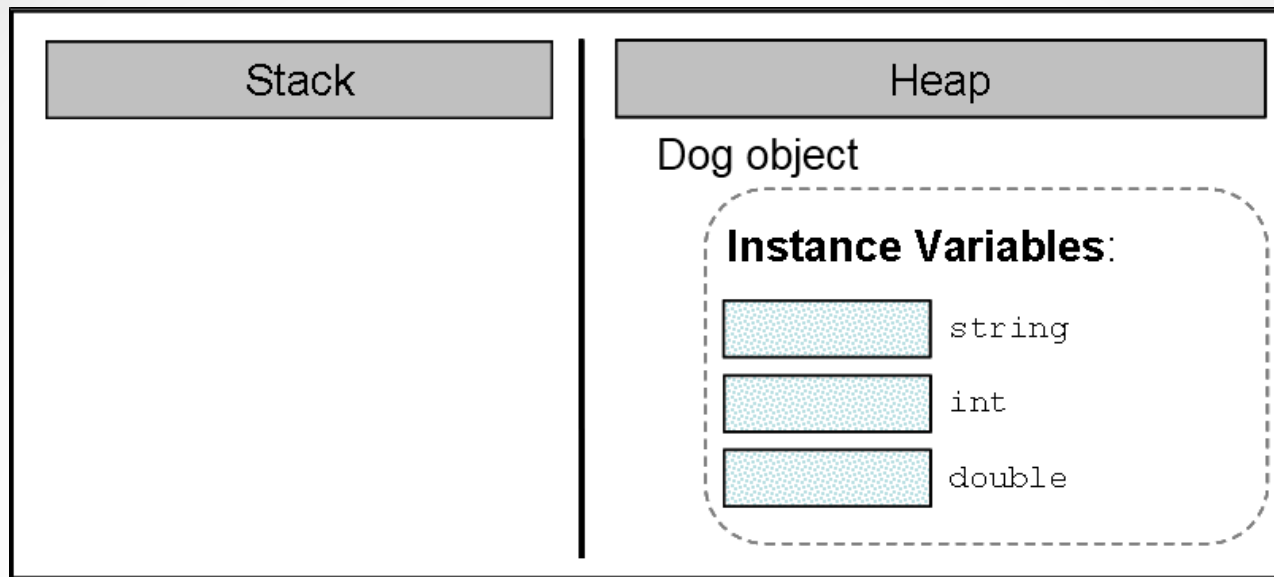
```

class inventaire
{
    public inventaire()
    {
        this.nom = "";
        this.nombreitem = -1;
        this.prix = 0.0;
        this.unitesEnStock = 0;
    }
    public inventaire(string nom)
    {
        this.nom = nom;
        this.nombreitem = -1;
        this.prix = 0.0;
        this.unitesEnStock = 0;
    }
    public inventaire(string nom, int nombreitem, double prix)
    {
        this.nom = nom;
        this.nombreitem = nombreitem;
        this.prix = prix;
        this.unitesEnStock = 0;
    }
    public inventaire(string nom, int nombreitem, double prix, int nombreenstock)
    {
        this.nom = nom;
        this.nombreitem = nombreitem;
        this.prix = prix;
        this.unitesEnStock = nombreenstock;
    }
    //Ajoutez les méthodes additionnelles ici
    private string nom;
    private int nombreitem;
    private double prix;
    private int unitesEnStock;
}

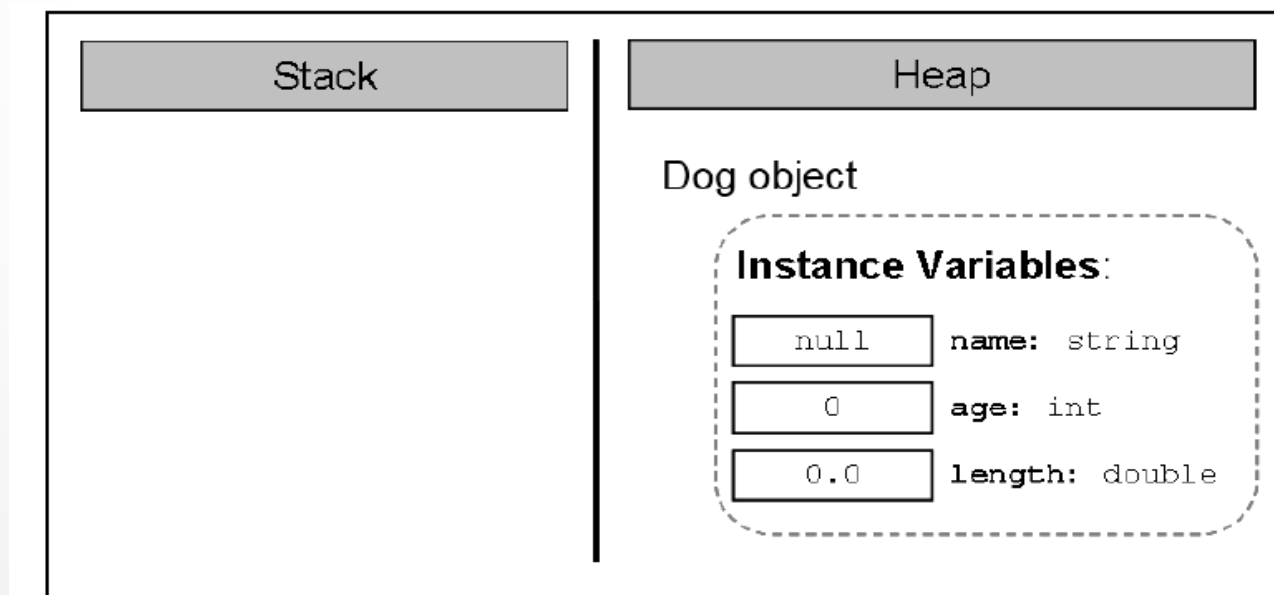
```

# Appel du constructeur

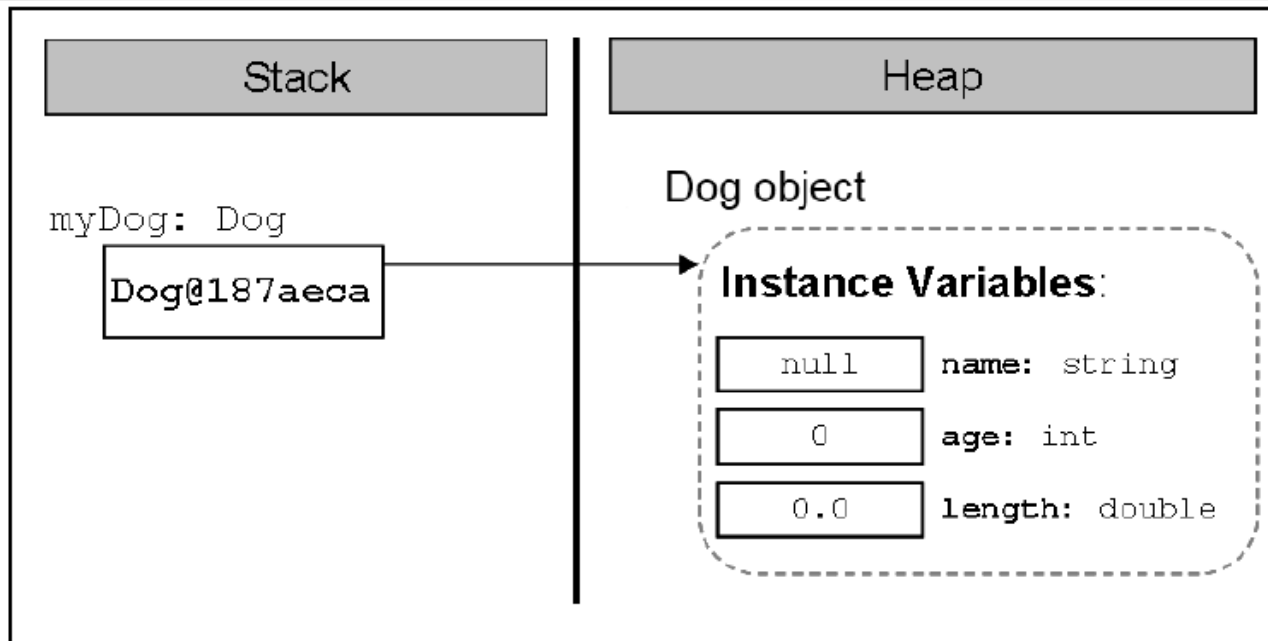
- Le processus d'appel du constructeur lors de la création d'un objet est le suivant:
  - Allocation de la mémoire à l'objet
  - Initialisation de champs (attributs)
  - Retour de la référence à l'objet.



Allocation de la mémoire



Initialisation des champs



Retour de la référence à l'objet