

# Programmation Orientée Objet

Java

Wijin – 2024



**JAVA**

**TP1**

## Cursus P00 – TP1

Créer une classe **Ville**, avec 3 attributs :

- « nomVille » (String)
- « nomPays » (String)
- « nombreHabitants » (int)

Créer une classe **Application**, avec une méthode main pour créer un objet de type **Ville**.

## Cursus P00 – TP1

Dans la classe **Ville**, ajouter deux méthodes **constructeurs** :

- Un premier sans paramètre
- Un second avec paramètres (initialisation des attributs)

Compléter la classe **Application** pour utiliser les constructeurs.

## Cursus P00 – TP1

Améliorer la classe **Ville** pour mettre en œuvre l'encapsulation :

- Attributs « private »
- Accesseurs

## Cursus P00 – TP1

Ajouter une méthode *afficheInfos* dans la classe **Ville** pour afficher les informations décrivant une ville (nom, pays et nombre d'habitants) :

- Cette méthode affiche une chaîne de caractères sur la sortie standard
- Cette méthode se base sur une méthode *toString* redéfinie dans la classe **Ville**

# Cursus P00 – TP1

Utilisation de la classe **Scanner** pour lire les entrées au clavier

La lecture des entrées clavier se fait via un objet de type **Scanner**.

La classe **Scanner** se trouve dans le package *java.util*.

Exemple :

```
import java.util.Scanner;
Scanner sc = new Scanner(System.in);
System.out.println("Veuillez saisir un mot :");
String str = sc.nextLine();
System.out.println("Vous avez saisi : " + str);
```

# Cursus P00 – TP1

Utilisation de la classe **Scanner** pour lire les entrées au clavier

Pour pouvoir récupérer ce qui est saisi dans la console, il faut initialiser l'objet de type **Scanner** avec l'entrée standard :

`System.in`

```
Scanner sc = new Scanner(System.in);
```

Il existe une méthode de récupération de données pour chaque type (sauf char) : *nextLine()* pour les String, *nextInt()* pour les int, ...

```
int i = sc.nextInt(); double d = sc.nextDouble();  
long l = sc.nextLong(); byte b = sc.nextByte();
```



## Cursus P00 – TP1

Utilisation de la classe **Scanner** pour lire les entrées au clavier

Modifier le constructeur sans paramètre de la classe **Ville** pour utiliser Scanner pour récupérer les valeurs à affecter sur les attributs

## Cursus P00 – TP1

Créer deux nouvelles classes héritant de **Ville** :

- **Capitale**
- **VillePortuaire**

La classe **Capitale** déclare un attribut « monument », de type String

La classe **VillePortuaire** déclare un attribut « activite », de type String

## Cursus P00 – TP1

Définir les constructeurs avec l'utilisation de **super**

Dans la classe **Application**, mettre en place la création de 2 villes, une capitale et une ville portuaire, et appeler la méthode « *afficheInfos* » sur chaque ville.

Faire les modifications nécessaires pour permettre d'afficher les informations spécifiques à la capitale et la ville portuaire.

## Cursus P00 – TP1

Proposer un code pour :

- Créer un tableau de villes
- Saisir au choix de l'utilisateur un code pour créer un objet de type **Capitale** ou **VillePortuaire** et le placer dans le tableau
- En fin de saisie, parcourir le tableau et afficher les infos de chaque ville