

# TP

## Développement Orienté Objet : Programmation objet/Java

---

### Exercice 1 :

Créer une classe appelée Employe pour modéliser un employé.

1. Cette classe doit avoir les attributs suivants :
  - nom (String) : Le nom de l'employé.
  - poste (String) : Le poste de l'employé (par exemple, "Manager", "Développeur").
  - salaire (double) : Le salaire de l'employé.
2. Implémenter un **constructeur par défaut** qui initialise :
  - le nom à "Inconnu",
  - le poste à "Non défini",
  - le salaire à 0.0.
3. Implémenter un **constructeur paramétré** qui permet de définir uniquement le nom et le poste, avec un salaire initialisé à 8000.0.
4. Implémenter un **autre constructeur paramétré** qui permet d'initialiser tous les attributs (nom, poste, salaire).
5. Ajoutez une **méthode** appelée afficherDetails() qui affiche les informations d' employé, comme le nom, le poste et le salaire.
6. Créer une classe principale appelée TestEmploye pour tester tous les constructeurs.

### Exercice 2 :

Créer un programme Java qui modélise des animaux en utilisant l'héritage. Voici les instructions :

1. Créer une **classe de base** appelée Animal avec :
  - Un attribut nom (String) pour stocker le nom de l'animal.
  - Un constructeur pour initialiser le nom.
  - Une méthode manger() qui affiche : "L'animal [nom] mange.".
2. Créer une **classe dérivée** appelée Chien qui hérite d'Animal :
  - Ajoutez un **constructeur** pour initialiser l'attribut nom.
  - Ajouter une méthode aboyer() qui affiche : "Le chien [nom] aboie : Woof Woof!".

- Redéfinir la méthode manger() pour afficher : "Le chien [nom] mange de la viande."
3. Créer une **autre classe dérivée** appelée Chat qui hérite d'Animal :
    - Ajoutez un **constructeur** pour initialiser l'attribut nom.
    - Ajouter une méthode miauler() qui affiche : "Le chat [nom] miaule : Meow!".
    - Redéfinir la méthode manger() pour afficher : "Le chat [nom] mange des croquettes."
  4. Créer une classe principale appelée TestAnimal pour :
    - Créer des objets de type Chien et Chat.
    - Appeler les méthodes manger(), aboyer() et miauler() sur les objets correspondants.

### Exercice 3

1. Créez une classe Livre qui doit contenir trois **attributs** :
  - titre : pour stocker le titre du livre.
  - auteur : pour stocker le nom de l'auteur.
  - prix : pour indiquer le prix du livre (un nombre).
  - Ajoutez un **constructeur** pour permettre d'initialiser ces trois attributs quand on crée un objet.
  - Ajoutez une **méthode** appelée afficherDetails() qui affiche les informations du livre, comme le titre, l'auteur et le prix.
2. Créez une autre classe appelée LivreTechnique, qui hérite de la classe Livre.
  - Ajoutez un nouvel **attribut** appelé domaine pour indiquer le domaine technique du livre (comme "Informatique").
  - Ajoutez un constructeur pour initialiser à la fois les attributs de la classe Livre (titre, auteur, prix) et l'attribut domaine.
  - Modifiez la méthode afficherDetails() pour inclure le domaine en plus des autres informations.
3. Créez une classe principale appelée TestLivre.
  - Dans cette classe, créez un **objet** de la classe Livre avec des valeurs comme titre, auteur et prix.
  - Ensuite, créez un autre **objet** de la classe LivreTechnique en ajoutant aussi une valeur pour le domaine.
  - Appelez la méthode afficherDetails() sur chaque objet pour vérifier que les informations s'affichent correctement.

#### Exercice 4 :

1. Créez une classe `Personne` avec les informations de base d'une personne :
  - **Attributs** : nom, age.
  - Un **constructeur** pour initialiser ces attributs.
  - Une méthode `afficherDetails()` qui affiche les informations générales d'une personne.
2. Créez une sous-classe `Etudiant` qui hérite de la classe `Personne`.
  - Ajoutez un nouvel attribut `numeroEtudiant` (le numéro de l'étudiant).
  - Ajoutez un constructeur pour initialiser à la fois les attributs de la classe `Etudiant` (nom, age) et l'attribut `numeroEtudiant`.
  - Redéfinissez la méthode `afficherDetails()` pour afficher les informations d'un étudiant, y compris son numéro.
3. Créez une sous-classe `EtudiantEtranger` qui hérite de la classe `Etudiant`.
  - Ajouter un attribut supplémentaire `paysOrigine` (le pays d'origine de l'étudiant).
  - Ajoutez un constructeur pour initialiser à la fois les attributs de la classe `EtudiantEtranger` (nom, age, `numeroEtudiant`) et l'attribut `paysOrigine`.
  - Redéfinissez la méthode `afficherDetails()` pour inclure le pays d'origine dans les informations affichées.
4. Créez une classe principale appelée `TestPersonne`
  - Dans cette classe créer un objet de chaque classe (`Personne`, `Etudiant`, et `EtudiantEtranger`).
  - Appeler la méthode `afficherDetails()` sur chaque objet pour observer les résultats.