# 📘 Devoir Python – Programmation Orientée Objet et Structures de Données

Ce devoir contient une série d'exercices variés (simples et complexes) sur la Programmation Orientée Objet (POO), les listes, dictionnaires et tuples. Certains exercices nécessitent de combiner plusieurs notions. Travaille bien chaque question en analysant bien l’énoncé. Bon courage ! 💡

## Exercice 1 [Simple] – Création d’une classe

Crée une classe `Chien` avec les attributs `nom`, `race`, et `âge`. Ajoute une méthode `aboyer()` qui affiche "Woof!".

## Exercice 2 [Complexe] – Classe avec méthode et dictionnaire

Crée une classe `Etudiant` avec `nom`, `niveau`, et un dictionnaire `notes`. Ajoute une méthode pour afficher la moyenne des notes.

## Exercice 3 [Complexe] – Liste d’objets

Crée une classe `Livre` avec `titre`, `auteur`, `année`. Crée une liste contenant plusieurs livres, puis affiche les livres publiés après 2015.

## Exercice 4 [Simple] – Utilisation de tuples

Crée un tuple contenant les coordonnées GPS (latitude, longitude). Affiche-les.

## Exercice 5 [Complexe] – Tuples dans une classe

Crée une classe `Lieu` avec `nom` et `coordonnées` (un tuple latitude, longitude). Crée 3 objets et affiche la distance approximative entre eux (simulation avec différence absolue).

## Exercice 6 [Complexe] – Classe avec dictionnaire dynamique

Crée une classe `Panier` avec un dictionnaire `produits` (clé = nom du produit, valeur = quantité). Ajoute des méthodes pour : ajouter un produit, supprimer un produit, afficher le panier.

## Exercice 7 [Très complexe] – Système de gestion d'étudiants

Crée un système avec une classe `Etudiant` (nom, prénom, notes), et une classe `Classe` qui contient une liste d’étudiants. Ajoute des méthodes dans `Classe` pour ajouter un étudiant, afficher tous les étudiants, et calculer la moyenne générale de la classe.

## Exercice 8 [Très complexe] – Base de données simple avec dictionnaire

Crée une classe `Contact` avec `nom`, `téléphone`, `email`. Crée une classe `Annuaire` contenant un dictionnaire de contacts (clé = nom). Ajoute des méthodes pour : ajouter un contact, rechercher par nom, supprimer un contact, et afficher tous les contacts.

## Exercice 9 [Simple] – Affichage de données structurées

Crée une liste contenant des tuples (nom, âge). Parcours cette liste pour afficher "Nom : X, Âge : Y" pour chaque personne.