# **Exercices récapitulatifs sur les bases de Python**

 **Niveau 1**

1. Demander le prénom de l’utilisateur et afficher "Bonjour <prénom> !".
2. Demander un nombre avec input, puis afficher son double en utilisant int() et float().
3. Demander l’âge et afficher "Majeur" ou "Mineur".
4. Vérifier si un nombre est compris entre 10 et 20 inclus.
5. Vérifier si un nombre est pair ou impair avec un if en une ligne.
6. Demander une chaîne et afficher sa première et dernière lettre.
7. Afficher un nombre avec 2 décimales et centré sur 10 caractères.
8. Demander un mot de passe jusqu’à ce que l’utilisateur tape "Python".
9. Afficher tous les nombres de 1 à 15.
10. Créer une liste avec les nombres de 1 à 10 et afficher la somme.
11. : Supprimer les doublons d’une liste [1,2,2,3,3,3,4].
12. Créer un dictionnaire {"nom": ..., "age": ...} et afficher les valeurs.

**Niveau 2**



1. : Créer saluer(prenom, message="Bonjour") et tester avec et sans le message.
2. Écrire une fonction somme(\*args) qui additionne tous les nombres passés en argument.
3. Créer une fonction infos\_personne(\*\*kwargs) et afficher clé : valeur pour chaque élément.
4. Créer une liste des carrés des nombres pairs de 0 à 20.
5. Parcourir un dictionnaire notes = {"Alice": 15, "Bob": 12, "Charlie": 18} et afficher Alice : 15 etc.
6. Demander une commande (start, stop, pause) et afficher une action correspondante.

**🔹 Niveau 3**



1. Écris une fonction qui, à partir d’une liste de mots, regroupe ceux qui sont des anagrammes.

*Exemple :*

*["écouter", "couteer", "route", "toure", "chien", "niche"]*

*Donnera :*

*["écouter", "couteer"],*

*["route", "toure"],*

*["chien", "niche"]*

*Tips : utiliser un dict avec les lettres triées comme clé, et accumuler les mots dans des listes.*

1. Écris un programme qui vérifie si une grille de Sudoku (9×9) est valide :

* Chaque ligne contient les chiffres 1 à 9 sans doublon.
* Chaque colonne idem.
* Chaque sous-grille 3×3 idem.

1. Écris une fonction qui compresse une chaîne en comptant les occurrences consécutives des caractères. Exemple :

"aaabbcdddddd" devient "a3b2c1d6"

Et écris aussi la fonction inverse pour décompresser.