

# TD - Bases de Python

ali.zainoul.az@gmail.com

## 1. Affichage, saisie et f-strings

**Objectif :** Maîtriser les fonctions `print()`, `input()` et l'affichage formaté.

**Exemple :**

```
name = input("Enter your full name: ")
age = int(input("Enter your age: "))
print(f"{name.title()} will be {age + 1} years old next year.")
```

**Exercice 1 :** Écrivez un programme qui demande un prénom, un nom et une année de naissance, puis affiche une phrase complète contenant toutes les informations de façon formatée et justifiée sur 40 caractères.

**Exercice 2 :** Demandez à l'utilisateur une température en Celsius et affichez l'équivalent en Fahrenheit avec 2 chiffres après la virgule.

## 2. Types fondamentaux et booléens

**Objectif :** Comprendre les types primitifs de Python et manipuler les booléens.

**Exemple :**

```
value = -3.5
is_positive = value > 0
print(f"The value is positive: {is_positive}")
```

**Exercice 3 :** Écrire un programme qui demande à l'utilisateur deux valeurs et affiche si elles sont de même type, si elles sont numériques, et si l'une est supérieure à l'autre.

**Exercice 4 :** Implémentez un test logique combiné avec des opérateurs `and`, `or`, `not` pour vérifier si une valeur est comprise entre deux bornes, mais pas égale aux bornes.

## 3. Nombres : int, float, complex

**Objectif :** Manipuler les différents types numériques de Python.

**Exemple :**

```
a = 5
b = 2.0
c = complex(a, b)
print(f"The complex number is: {c}")
```

**Exercice 5 :** Demandez deux nombres à l'utilisateur et affichez leur quotient, reste, puissance, racine carrée de leur somme et leur conversion en nombre complexe.

**Exercice 6 :** Implémentez un convertisseur d'un nombre réel vers sa notation scientifique avec 3 décimales, puis affichez sa partie entière et fractionnaire.

## 4. Chaînes de caractères

**Objectif :** Définir et manipuler les chaînes de caractères.

**Exemple :**

```
s = '''This is a multi-line
string with "quotes" and 'apostrophes'.'''
print(s)
```

**Exercice 7 :** Définissez une chaîne multi-lignes contenant une citation littéraire, affichez-la en mode `raw`, puis comptez le nombre de mots et la fréquence d'un mot donné.

**Exercice 8 :** Écrivez une fonction qui reçoit une chaîne et retourne une nouvelle version :

- sans espaces en début/fin,
- tout en minuscules,
- sans voyelles.

## 5. Indijage, slices, opérateurs sur les chaînes

**Objectif :** Accéder et manipuler les caractères d'une chaîne.

**Exemple :**

```
text = "Hello, World!"
print(text[-1])          # last character
print(text[0:5])         # Hello
print("World" in text)
```

**Exercice 9 :** Créez une fonction qui prend une chaîne et retourne la chaîne inversée. Puis utilisez cette fonction pour tester si une chaîne est un palindrome (même à l'envers).

**Exercice 10 :** Utilisez une boucle pour afficher en diagonale les lettres d'une chaîne, puis reconstruisez la chaîne en supprimant les lettres en position impaire (en utilisant l'indijage).

## 6. Méthodes essentielles et affichage avancé

**Objectif :** Découvrir les méthodes clés des chaînes et enrichir l'affichage.

**Exemple :**

```
s = " Python is Fun "
print(s.strip().upper().replace("FUN", "powerful"))
```

**Exercice 11 :** Demandez une phrase à l'utilisateur. Affichez :

- le nombre de mots,
- le mot le plus long,

- les mots triés par ordre alphabétique inversé,
- et la phrase avec chaque mot capitalisé.

**Exercice 12 :** Implémentez un tableau d’affichage formaté de plusieurs personnes (nom, âge, métier) à l’aide de **f-strings** avec alignement et largeur fixe.