# Premiers pas en python

Objectifs :

- Comprendre ce qu'est Python.
- Savoir écrire et exécuter un premier programme.
- Déclarer une variable, afficher son contenu, identifier son type..
- Réaliser des opérations simples avec des variables.

## Table des matières

I. Découverte de Python, affichage, variables....1
1. Qu'est-ce qu'un programme ? qu'est-ce que python ?....1
Présentation rapide....1
Premier code : print(" Hello World ")....2
Notions de variables....2
Opérations simples....3
Demande d'information à l'utilisateur : la fonction input()....3
Petits exercices....3
2. Les variables sont typées....4
Identification des types....4
Conversions de types....4
Expérimentation libre....4
Questions de synthèse :....5
3. Bilan....5
II. Exercice supplémentaire (devoir)....5

# I. Découverte de Python, affichage, variables.

## 1. Qu'est-ce qu'un programme ? qu'est-ce que python ?

### Présentation rapide

- Python = langage **simple**, **lisible**, utilisé pour **électronique, réseaux, scripts, IA...**
- Programme = fichier texte → interprété par Python.
- Nécessité d'utiliser un IDE. Il en existe beaucoup, pour les premières séances nous utiliserons Thonny.

**Phrase-clé** :

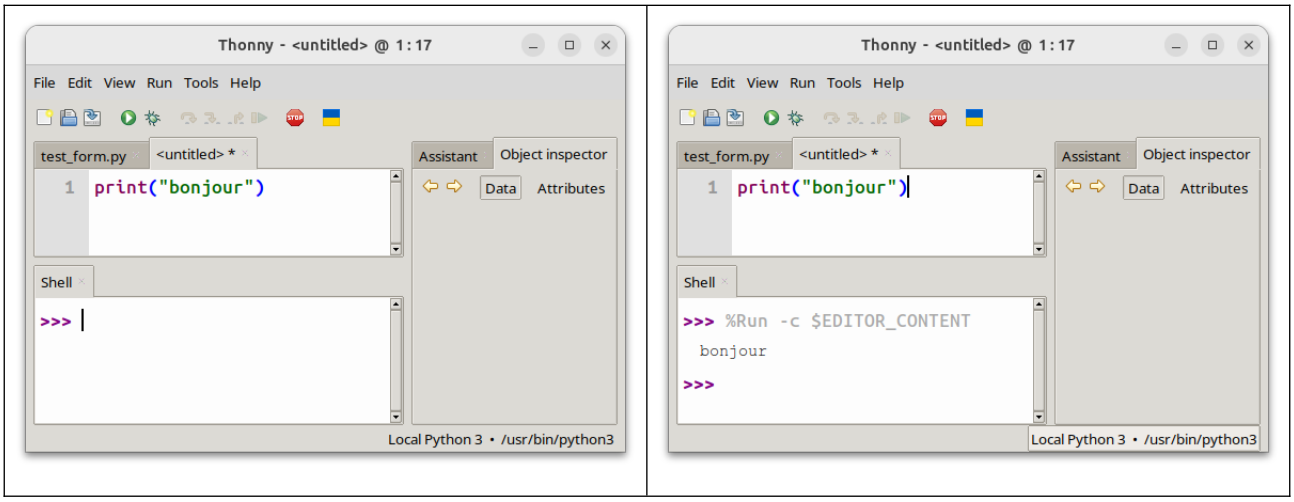
*"Un programme Python, c'est juste une série d'instructions que l'ordinateur suit, ligne par ligne."*

## Premier code : print(" Hello World ")

Ouvrez Thonny puis ouvrez un nouveau fichier et taper le texte ci-dessous dans la fenêtre supérieure , la fenêtre d'écriture du programme :

```
print("bonjour")
```

☑ Appuyer sur l'icone représentant une flèche vers la droite (touche « play ») pour lancer l'exécution du programme. Le résultat de cette exécution devrait normalement s'afficher dans la fenêtre d'exécution dans la partie basse de Thonny.




L'instruction print() commande l'affichage d'une chaîne de caractères passée en paramètre de la fonction. Ici la chaîne de caractère est « bonjour ».

## Notions de variables

- **Variable** = étiquette vers le lieu de stockage d'une valeur.

**Exemple de programme simple utilisant les variables** :

```
nom = "Alice"
âge = 18
température = 23.5

print(nom)
print(âge)
print(température)
```

☑ Exécuter le programme précédent. Les trois premières instructions sont des opérations d'affectation. Chacune affecte une valeur à une variable. La première instruction affecte la valeur *« Alice »* à la variable *nom*. La deuxième instruction affecte la valeur *18* à la variable *âge*. La première instruction *print()* commande l'affichage de la variable *nom* ; la seconde instruction *print()* affiche le contenu de la variable *âge*, la troisième instruction *print()* affiche la valeur de la variable *température*.

## Opérations simples

- Additionner deux nombres.
- Multiplier, soustraire, diviser.

```
a = 5
b = 3
somme = a + b   # déclaration de la variable somme
produit = a * b
print("La somme est :", somme)
print("Le produit est :", produit)
```

Taper et exécuter le programme ci-dessus. La troisième ligne du programme contient l'instruction :
`somme = a + b`
le reste de la ligne, le caractère # et tout ce qui le suit juqu'à la fin de la ligne, est ignoré au moment de l'exécution du programme. Il s'agit d'un commentaire, c'est à dire d'un texte placé dans le programme pour aider le programmeur.

## Demande d'information à l'utilisateur : la fonction input()

objectif : :

- Demander à l'utilisateur son prénom (`input()`),
- Afficher un message de bienvenue.

```
prenom = input("Quel est ton prénom ? ")
print("Bienvenue", prenom)
```

## Petits exercices

**Exercice 1 :**
Créer un programme qui :

- Déclare une variable `longueur` à laquelle on affecte la valeur 4.5 ,
- Déclare une variable `largeur` à laquelle on affecte la valeur 2.0 ,
- Calcule l'aire d'un rectangle,
- Affiche le résultat.

# 2. Les variables sont typées

En Python, chaque donnée possède un **type** qui détermine ce que l'on peut faire avec elle. Par exemple, on additionne deux nombres, on concatène deux chaînes de caractères, etc.
Les types les plus courants sont :

- `int` : entier
- `float` : nombre décimal
- `str` : chaîne de caractères
- `bool` : booléen (`True` ou `False`)

## Identification des types : fonction type()

Le type d'une variable est accessible par la fonction type() ; que l'on peut afficher avec print()

Exemple :

```
print( type(5) )

<class 'int'>
```

Complèter le tableau suivant en indiquant le type de chaque expression en Python.

| Expression Python | Valeur obtenue | Type |
|---|---|---|
| `5 + 2` | | |
| `3.0 * 2` | | |
| `"5" + "2"` | | |
| `type(7.5)` | | |
| `10 > 3` | | |
| `bool("Python")` | | |
| `int("42")` | | |

## Conversions de types

Donne ce que renvoie chaque ligne de code, puis teste-la dans Python pour vérifier.

```
int(3.14)
float("5")
str(True)
bool(0)
bool("")
```

**Question bonus** : Pourquoi `bool("")` renvoie-t-il `False`, alors que `bool("0")` renvoie `True` ? (toute chaîne non vide est évaluée à True)

## Expérimentation libre

**Consigne** : À partir de ce que tu as appris, propose 3 expressions différentes en mélangeant les types (`int`, `float`, `str`, `bool`) et indique ce que tu attends comme résultat.

Exemple :

```
"Résultat : " + str(4 * 5)
```

**Exercice 2 :**

- Demander deux nombres à l'utilisateur,
- Afficher leur somme.

**Exercice 3 :**

Utilisation des opérateurs // (division entière) et % (modulo = reste de la division entière) ;

- Demander deux nombres a et b à l'utilisateur,
- Afficher proprement le résultat de la division entière de a par b et le reste de cette division entière de a par b.

### Questions de synthèse :

1. Quelle est la différence entre `5` et `"5"` en Python ?
2. Quelle conversion est nécessaire pour additionner un entier et une chaîne contenant un chiffre ? (comment additionner 12 et `"7"` par exemple)
3. Pourquoi `bool("False")` renvoie-t-il `True` ?

## 3. Bilan

À la fin de la séance, ils doivent être capables de :

- Écrire un script de 5-10 lignes,
- Affecter une valeur à une variable ;
- Afficher du texte et des nombres ;
- Demander une valeur et la manipuler ;
- Identifier et comprendre l'usage d'un commentaire
- Comprendre la notion de type de données

**Mots-clés à retenir** : print, variable, input, opérateurs (+, -, *, /), #…
**Compétence validée** : Écrire un premier script Python sans erreur de syntaxe.

# II. Exercice supplémentaire (devoir)

**Titre** : Calculateur de périmètre et d'aire d'un rectangle

**Consignes** :

Écris un programme Python qui :

- Demande à l'utilisateur de saisir la **longueur** et la **largeur** d'un rectangle (nombres à virgule possibles).
- Calcule :
  - Le périmètre du rectangle : Périmètre=2×(longueur+largeur)
  - L'aire du rectangle : Aire=longueur×largeur
- Affiche les deux résultats de manière claire.

**Exemple d'exécution attendue** :

```
Entrez la longueur : 5.0
Entrez la largeur : 3.0
Le périmètre du rectangle est : 16.0
L'aire du rectangle est : 15.0
```