

# Afficher ses variables avec print

La fonction `print()` est une des bases de la programmation en Python. Elle permet d'afficher du texte, des nombres ou des résultats sur l'écran, ce qui est essentiel pour communiquer avec l'utilisateur ou vérifier le fonctionnement d'un programme.

5GMS

6GMS

4GMS

3TTR



## Objectif

Apprendre à utiliser `print()` pour afficher différentes informations de manière claire et organisée sur l'écran.

## Introduction à `print()`

La fonction `print()` permet d'afficher du contenu dans le terminal. Sa syntaxe de base est la suivante :

```
1 | print(valeur)
```

(Attention aux parenthèses)

Exemple :

```
1 | print("Bonjour tout le monde !")
```

Résultat :

Bonjour tout le monde !

## Afficher plusieurs valeurs

On peut afficher plusieurs valeurs en les séparant par des virgules. Par défaut, `print()` ajoute un espace entre chaque valeur.

Exemple :

```
1 | print("Bonjour", "tout", "le", "monde", "!")
```

Résultat :

Bonjour tout le monde !

---

## Affichage de variables

On peut également utiliser `print()` pour afficher le contenu de variables.

Exemples :

```
1 | annee = 2025
2 | print(annee) # affiche: 2025
3 | print("Nous sommes en", annee) # affiche: Nous sommes en 2025
```

```
1 | nom = "Alice"
2 | age = 25
3 | print("Nom :", nom, "Âge :", age)
```

Résultat :

Nom : Alice Âge : 25

---

## Changer le séparateur entre les valeurs

Par défaut, les valeurs affichées avec `print()` sont **séparées par un espace** (" "). On peut modifier cela grâce au paramètre `sep`.

Exemple :

```
1 | print("Python", "est", "génial", sep="-")
```

Résultat :

Python-est-génial

---

## Modifier la fin de ligne

Par défaut, `print()` termine chaque affichage par un saut de ligne (`\n`). On peut changer cela grâce au paramètre `end`.

### Exemple :

```
1 | print("Bonjour", end=" ")
2 | print("tout le monde !")
```

Résultat :

Bonjour tout le monde !

---

## À retenir

---

- La fonction `print()` est utilisée pour afficher du contenu.
- Elle peut afficher des chaînes, des nombres, des variables, ou des calculs.
- Les paramètres `sep` et `end` permettent de personnaliser l'affichage.

Avec une bonne maîtrise de `print()`, tu peux rendre tes programmes interactifs et clairs pour l'utilisateur ou pour toi-même !