

# Manuel 3 - Fonctions Python

Inspiré du cours de Michaël Belheur

25 janvier 2024

## 1 Introduction

Les fonctions en Python sont des blocs de code organisés et réutilisables qui sont utilisés pour effectuer une action spécifique. Elles permettent de segmenter des programmes complexes en petites parties gérables et réutilisables.

```
1 # Exemple simple de fonction
2 def saluer():
3     print("Bonjour!")
4 saluer()
```

## 2 Définition et Utilisation des Fonctions

### 2.1 Structure de Base d'une Fonction

Une fonction est définie avec le mot-clé `def` suivi du nom de la fonction, des parenthèses contenant éventuellement des paramètres et d'un bloc de code indenté.

```
1 def nom_de_la_fonction(param1, param2):
2     # Bloc de code
3     return resultat
4
5 # Structure de base d'une fonction
6 def ma_fonction():
7     print("Ceci est une fonction.")
8 ma_fonction()
```

### 2.2 Paramètres et Arguments

Les paramètres sont des variables spécifiées dans la définition de la fonction. Les arguments sont les valeurs réelles fournies lors de l'appel de la fonction.

```
1 # Fonction avec paramètres
2 def saluer_personne(nom):
3     print("Bonjour " + nom)
4 saluer_personne("Alice")
```

## 2.3 Appel de Fonctions

Pour exécuter une fonction, il suffit de l'appeler par son nom suivi de parenthèses contenant les arguments.

```

1 resultat = nom_de_la_fonction(valeur1, valeur2)

1 # Appel de fonction
2 def additionner(a, b):
3     return a + b
4 resultat = additionner(5, 3)
5 print(resultat)
```

## 3 Types de Fonctions

### 3.1 Fonctions sans Retour

Certaines fonctions ne retournent pas de valeur. Elles sont utilisées pour leur effet.

```

1 def afficher_message():
2     print("Bonjour tout le monde!")

1 # Fonction sans valeur de retour
2 def afficher_message():
3     print("Ceci est un message.")
4 afficher_message()
```

### 3.2 Fonctions avec Retour

Ces fonctions renvoient une valeur. Elles utilisent l'instruction `return`.

```

1 def addition(a, b):
2     return a + b

1 # Fonction avec valeur de retour
2 def multiplier(a, b):
3     return a * b
4 produit = multiplier(2, 3)
5 print(produit)
```

## 4 Portée des Variables

### 4.1 Variables Locales

Les variables définies à l'intérieur d'une fonction sont locales à cette fonction.

```

1 # Utilisation de variables locales
2 def ma_fonction_locale():
3     x = 10 # variable locale
4     print(x)
5 ma_fonction_locale()
```

## 4.2 Variables Globales

Les variables définies à l'extérieur de toutes les fonctions sont globales.

```
1 # Utilisation de variables globales
2 x = 10 # variable globale
3 def ma_fonction_globale():
4     print(x)
5 ma_fonction_globale()
```