



PYTHON : FICHE RÉCAP

LES FONCTIONS

Les fonctions vous permettent d'éviter de recopier/coller des mêmes parties de code à plusieurs endroits de votre programme, et donc de faire du code plus synthétique et plus maintenable. En savoir plus sur les fonctions :

<https://www.youtube.com/watch?v=1NUkHC-Fzx4>

Créer et appeler une fonction

```
def afficher_nom():  
    print("Je m'appelle")  
    print("Toto")
```

Pour créer une fonction on utilise "def" pour "définir" la fonction

afficher_nom() # Appel de la fonction

Attention : Il faut définir la fonction avant de l'appeler

Paramètres

```
def afficher_nom(nom):  
    print("Je m'appelle")  
    print(nom)
```

```
afficher_nom("Toto")  
afficher_nom("Titi")
```

Ici le paramètre nom est obligatoire, on est obligé de passer une valeur à l'appel de la fonction

```
def afficher_nom(nom = ""):
    if nom == "":
        print("Le nom est vide")
        return
    print("Je m'appelle")
    print(nom)
```

Ici le paramètre nom est optionnel, on peut appeler la fonction sans passer le paramètre si on le souhaite :

```
afficher_nom()
```

Return

Return permet soit de retourner des valeurs de sortie de la fonction, soit de sortir à tout moment de la fonction.

Il est inutile (et même déconseillé) de mettre return à la fin d'une fonction si vous ne retournez aucune valeur.

```
def demander_nom():
    nom = input("Quel est votre nom: ")
    return nom
```

```
nom_personne = demander_nom()
```

Ici la fonction retourne la valeur du nom reçu de input (la fonction input retourne elle aussi une valeur)

Fonction récursive

Une fonction peut se rappeler elle-même. Dans ce cas elle devient récursive.

Attention, la récursivité a toujours "une condition de sortie", sinon on va boucler de manière infinie.

Ici la condition de récursion est que le nom est vide, donc il faut réessayer.

La condition de sortie est que le nom n'est pas vide, et donc on retourne la valeur du nom

```
def demander_nom():  
    nom = input("Quel est votre nom: ")  
    if nom == "":  
        print("ERREUR : Le nom ne doit pas être vide")  
        return demander_nom()  
    return nom
```

```
nom_personne = demander_nom()
```

Variables globales et locales

Les variables créées dans une fonction (et les paramètres) sont 'locales à la fonction', c'est à dire qu'on ne peut les utiliser que à l'intérieur de la fonction.

Ici on ne pourrait pas faire un print(a) au niveau du programme principal. Car 'a' est une variable locale à la fonction afficher_nom.

Par contre la variable b a été définie dans le programme principal. 'global b' permet à la fonction de venir altérer la valeur 'b' du programme principal au lieu de recréer une variable 'b' locale à la fonction

```
b = 10
```

```
def afficher_nom(nom = ""):  
    global b  
    b = 15  
    a = 5  
    if nom == "":  
        print("Le nom est vide")  
        return  
    print("Je m'appelle")  
    print(nom)  
    print(b)
```

```
print(b)
```