



1. Les fonctions

1.1. Définition d'une fonction

Il est parfois utile d'isoler une partie d'un programme pour pouvoir l'utiliser à n'importe quel moment et autant de fois que souhaité. Il s'agit de sous-programme appelé « **fonctions** ». Ces fonctions facilitent l'écriture d'un programme en le rendant moins long et plus lisible.

Une **fonction** est un ensemble d'instructions réutilisables. L'instruction Python « **def** » suivi du nom de la fonction permet de la définir.

Une fonction peut prendre des valeurs en entrée, celles-ci sont utilisées dans des variables appelées « **paramètres** ». Pour exécuter la fonction, il est nécessaire de préciser les valeurs de ces variables. Lors de la définition d'une fonction, les paramètres se mettent entre parenthèses après son nom.

Exemple : fonction nommée `bonjour`, qui permet d'afficher « Bonjour ! »

```
def bonjour() :  
    print (« Bonjour ! »)
```

Exemple fonction nommée `bonjour` ayant un paramètre nommé « prénom »

```
def bonjour(prenom) :  
    print (« Bonjour »,prenom )
```

1.2. Le résultat d'une fonction

Une fonction renvoie une valeur (nombre, texte...) grâce à l'instruction « **return** ». Pour cela, il faut écrire le mot `return` suivi d'une variable de sortie.

Remarque : il est possible de ne rien renvoyer en sortie. Par exemple, lorsque la fonction permet un simple affichage.

Exemple : fonction nommée `produit` qui renvoie le produit de deux nombres :

```
def produit(a,b)  
    return (a*b)
```

1.3. L'appel d'une fonction

Dans le programme principal, on appelle une **fonction** en écrivant son nom suivi des valeurs des paramètres entre parenthèses.

Exemple : Ecriture d'un programme utilisant deux appels à une même fonction nommée « `tempsTrajet` »

```
def tempsTrajet (distance, vitesse) :  
    return distance/vitesse  
  
temps1=tempsTrajet(200,110)  
print(temps1)  
  
temps2=tempsTrajet(150,80)  
print(temps2)
```

Une fonction peut être appelée à plusieurs reprises dans un programme, avec des valeurs de paramètre identiques ou non. L'ordre des paramètres doit être respecté dans l'appel de la fonction