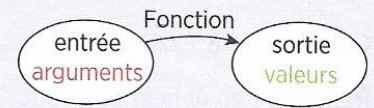


TP ALGO 3 : LES FONCTIONS

En général, une fonction informatique a une entrée (des informations qu'on lui fournit appelées arguments ou paramètres), exécute un travail sur ces arguments et renvoie des informations en sortie, comme des nombres et/ou des messages.



Syntaxe :

On définit une telle fonction en Python (dans la console ou dans un script) par :

```
def nom de la fonction (argument1, argument 2, ...) :  
    instructions éventuelles  
    return résultat1, résultat2, ...
```

Commentaire

return *résultat1*,... a deux effets :

- renvoyer le(s) résultat(s) obtenu(s)
- terminer l'exécution des instructions définissant la fonction.

Il existe aussi des fonctions sans argument.

Exemple 1 : une fonction « comme en math »

- La fonction Python `f` définie ci-contre correspond à la fonction $x \mapsto 3x + 2$.
- L'exécution de `f(4)` renvoie la valeur de $3 \times 4 + 2$ pour x valant 4, c'est-à-dire 14. On dit que l'appel `f(4)` **renvoie 14**.
- On peut réutiliser l'écriture `f(4)` dans d'autres calculs, comme en math.

```
def f(x):  
    return 3*x + 2
```

```
>>> f(4)  
14
```

```
>>> f(4) + 6  
25
```

Remarque : il existe des fonctions qui ne renvoient rien ; elles peuvent par exemple afficher un résultat sans qu'il soit réutilisable dans une autre instruction.

Exemple 2 :

La fonction Python définie ci-contre a pour nom `carres` et pour arguments `a` et `b`.

Elle renvoie deux résultats.

Quand on exécute `carres(4,1)`, `diff` vaut $4^2 - 1^2$, soit 15 et `somme` vaut $4^2 + 1^2$, soit 17.

Donc `carres(4,1)` renvoie (15, 17).

```
def carres(a,b):  
    diff = a**2 - b**2  
    somme = a**2 + b**2  
    return diff, somme
```

```
>>> carres(4,1)  
(15,17)
```

Pourquoi utiliser des fonctions ?

Utiliser des fonctions permet d'effectuer des opérations répétitives sans devoir saisir les instructions à chaque fois. Cela permet également de découper un programme en parties plus simples et de rendre le programme plus lisible.

Exercices d'application

Exercice 1

On a défini une fonction Python par les instructions ci-contre.

```
def g(x):  
    return 2*x*(x + 4)
```

1. Quel est son nom ?
2. Combien a-t-elle d'arguments (ou de paramètres)
3. Que renvoient les appels :

$g(1)$

$g(2)$

$g(4)$

$3 * g(1) + 4$

Exercice 2

On veut définir une fonction Python nommée aire qui renvoie l'aire d'un triangle de base b et de hauteur h .

1. Combien a-t-elle d'arguments et lesquels ?
2. Quel résultat doit-elle renvoyer ?
3. Ecrire le script définissant la fonction aire. Le tester pour une base de 5 cm et une hauteur de 4 cm.

Exercice 3

1. Ecrire une fonction qui prend en paramètres deux nombres et renvoie le plus grand des deux.
2. Même question avec trois nombres.

Exercice 4

1. Ecrire une fonction valeur_absolue qui prend en paramètre un nombre et renvoie sa valeur absolue.
2. Ecrire une fonction distance qui prend en paramètre deux nombres et renvoie la distance entre ces deux nombres.

Exercice 5

Une société de location de vélos propose les tarifs suivants : 7,50 € de l'heure pour les 3 premières heures de location, puis 2 € par heure supplémentaire. Au-delà de 6 h de location, une réduction de 10 % est également appliquée.

1. Quel est le tarif pour 2 heures de location ? Pour 5 heures de location ? Pour 7 heures de location ?
2. Ecrire le script d'une fonction Python qui renvoie le prix à payer selon le nombre d'heures de location.