

# Cours SI6

Python – Les fonctions par l'exemple



python

- Uniquement des fonctions en Python
- Pas de procédure
- Si fonction pas return -> retourne None
- Définit par le mot clef def

```
>>> def fib(n): # suite de fibonacci jusqu'e n
...     a, b = 0, 1
...     while b < n:
...         print b,
...         a, b = b, a + b
...
>>> fib(100)
1 1 2 3 5 8 13 21 34 55 89
```

## Introduction

- Peuvent être définis par défaut
  - Optionnels à l'appel
- Paramètres ordonnés ou nommés

```
>>> def welcome(name, greeting='Hello', mark='!'):
...     print greeting, name, mark
...
>>> welcome('world')
Hello world !
>>> welcome('monde', 'Bonjour')
Bonjour monde !
>>> welcome('world', mark='...')
Hello world ...
```

## Paramètres d'une fonction

- Fait partie intégrante du code
- Commentaires standards avec # pour les remarques techniques
- Code auto documenté avec `"""` utilisable:
  - Par les ide
  - Par la commande help

```
>>> def dummy():  
...     """Cette fonction est du pipeau...  
...  
...     Cette fonction est vraiment du pipeau,  
...     elle ne fait absolument rien."""  
...     pass  
  
>>> help(dummy)  
Help on function dummy in module __main__:  
  
dummy()  
    Cette fonction est du pipeau...  
  
    Cette fonction est vraiment du pipeau,  
    elle ne fait absolument rien.
```

# Documentation

```
>>> def funct1(val):  
...     return val > 0  
...  
>>> filter(funct1, [1, -2, 3, -4, 5])  
[1, 3, 5]
```

- Filtre une séquence grâce à une fonction personnalisée
- Retourne une séquence qui contient les éléments « True » de la séquence initiale

## Les Séquences : filter

```
>>> def sum(x, y):  
...     return x + y  
...  
>>> map(sum, [1, 2, 3], [4, 5, 6])  
[5, 7, 9]
```

- Applique une fonction personnalisée sur 1 ou plrs séquences

## Les séquences: map

- Réduit une séquence en appliquant une fonction récursivement sur chacun des éléments de la séquence
- Troisième paramètre optionnel -> valeur initiale du calcul

```
>>> reduce(sum, [1, 2, 3, 4, 5])  
15  
>>> reduce(sum, [1, 2, 3, 4, 5], -5)  
10
```

## Les séquences : reduce

- Une fonction est vue comme un objet:
  - On peut déclarer une fonction dans une fonction
  - On peut assigner une fonction à une variable
  - Une fonction peut retourner une fonction
- Une fonction peut retourner plusieurs valeurs!!!

**Spécificités fonctions Python**



- En entrée: `dataIn = ["MEDASSI;Anthony;1978/06/12", "NAU DIER;Nans;1978/10/19"]`
- En sortie: `['amedassi19780612', 'nnaudier19781019']`
- createLogin:
  - Prend 3 paramètres : nom, prenom, ddn
  - Retourne le login en minuscule composé de:
    - 1<sup>ère</sup> lettre du prénom
    - Le nom sans espace
    - La ddn sous la forme yyymmdd
- createTabLogin:
  - Prend le tableau d'entrée en paramètre
  - Retourne le tableau de login

## Exo01 : Création Login

The image shows a presentation slide. It has a wide beige border. Inside this border is a large, solid grey rectangle that occupies most of the slide's area. In the bottom-left corner of the slide, the word "Fin" is written in a bold, orange font.

**Fin**

