

## 1. Le langage Python : présentation et caractéristiques

Python est un langage interprété.

Il possède de nombreuses extensions (calcul scientifique, tracé de graphiques,...).

Extension : librairie contenant des fonctions dédiées à une utilisation spécifique.

Python est particulièrement utilisé pour le calcul scientifique.

Il est très répandu dans les universités et les écoles d'ingénieur.

## 2. Deux manières de travailler en Python

Il existe deux manières bien différentes d'utiliser Python :

- taper des commandes dans l'interpréteur ;
- éditer un programme dans un éditeur de texte, l'enregistrer puis l'exécuter.

### 2.1. Utiliser Python dans l'interpréteur :

C'est un mode interactif dans lequel l'utilisateur lance l'interpréteur Python, puis saisi une par une des lignes en langage Python. L'interpréteur traite les lignes au fur et à mesure qu'elles sont saisies.

L'interpréteur est lancé grâce à la commande python3 (pour utiliser la version 3 de Python).

>>> représente l'invite de commande de l'interpréteur Python.

Exemple :

```

Terminal
File Edit View Search Terminal Help
loic@loic-ThinkPad-Edge ~ $ python3
Python 3.3.1 (default, Apr 17 2013, 22:30:32)
[GCC 4.7.3] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> x=5
>>> from math import *
>>> sqrt(x)
2.23606797749979
>>> y=sqrt(x)
>>> type(x)
<class 'int'>
>>> type(y)
<class 'float'>
>>> y**2
5.000000000000001
>>> alpha=pi/3
>>> sin(alpha)
0.8660254037844386
>>> cos(alpha)
0.5000000000000001
>>>

```

Signification de ces commandes :

Déclarer  $x = 5$   
 Importer les fonctions de la librairie math  
 Calculer  $y =$  racine carrée de  $x$   
 Afficher le type de la variable  $x$   
 → int : nombre entier  
 Afficher le type de la variable  $y$   
 → float : nombre réel  
 Calculer  $y$  puissance 2  
 Déclarer l'angle  $\alpha = \pi/3$   
 Calculer sinus de  $\alpha$   
 Calculer cosinus de  $\alpha$

On constate une erreur sur la dernière décimale !

### Avantages et inconvénients du travail dans l'interpréteur :

Rapide lorsque l'on a que quelques lignes à taper, mais peu adaptée aux calculs plus longs.

Si une modification est nécessaire, obligation de retaper toutes les lignes ou une partie.

## 2.2. Éditer puis exécuter un script :

Un script est un fichier texte contenant des lignes de programme pour un langage interprété. En langage Python, le script doit porter l'extension « py ».

Écrivons le code donné dans les exemples suivants à l'intérieur du fichier test1.py.

```
from math import sqrt

x = 5
y = sqrt(x)
print("La racine carrée de %d vaut %f" % (x, y))
```

Importer sqrt() dans la librairie math

Nom des variables insérées dans la chaîne de caractères

Insérer un entier

Insérer un réel

Résultat :

```
La racine carrée de 5 vaut 2.236068
```

On obtient un résultat avec six décimales.

Cette manière d'insérer des variables dans une chaîne de caractères est courante mais dépréciée. Les dernières versions de Python offrent une autre méthode, à l'aide de la fonction `format()` :

```
from math import sin, pi
n = 4
alpha = pi/n
print("sin(pi/{0}) = {1:.3}".format(n, sin(alpha)))
```

Variables à insérer dans la chaîne

Valeur de n

Valeur de sin(alpha) avec 3 décimales

`{0}` et `{1}` représentent respectivement la valeur de la première et de la deuxième variable.

Résultat :

```
cos(pi/4) = 0.707
```

La syntaxe peut être simplifiée, en omettant le rang de la variable :

```
print("sin(pi/{}) = {:.3}".format(n, sin(alpha)))
```

La fonction `str()` permet de convertir un nombre en chaîne de caractères.

```
x = 1.5
print("x au carré est égal à " + str(x**2))
```

L'opérateur `+` permet de concaténer (mettre bout à bout) deux chaînes de caractères.

Résultat :

```
x au carré est égal à 2.25
```

Avantages de l'utilisation d'un script :

- Script facilement modifiable ;
- Possibilité de conserver le script pour le réutiliser ;
- Possibilité de travailler avec plusieurs fichiers (programmation modulaire).