

A lire en premier : Exécuter un programme python

Dans le cas d'une utilisation très simple de Python, par exemple comme calculatrice, vous pouvez utiliser le **mode interactif** (on parle aussi de **ligne de commande**). A chaque fois, vous entrez les commandes directement dans la fenêtre Terminal de l'interpréteur Python sans les sauvegarder au préalable dans un fichier. Cette méthode utile pour s'initier aux commandes de bases du langage présente toutefois un gros inconvénient : toutes les séquences d'instruction que vous avez écrites disparaissent irrémédiablement dès que vous fermez l'interpréteur.

C'est pourquoi vos programmes en Python (on parle aussi de **script Python**) devront être écrits dans l'éditeur de texte de l'IDE Python (Integrated Development Environment) disponible sur votre ordinateur puis sauves dans votre répertoire personnel sous la forme `nom-fichier.py`, `.py` est l'extension universellement utilisée pour reconnaître un script Python.

On peut utiliser l'éditeur de texte incorporé dans l'interface de développement **IDLE** (Python's Integrated DeveLopment Environment) qui fait partie de la distribution de Python. Il existe d'autres IDE plus sophistiqués et conviviaux tels que SPYDER et CANOPY.

Pour exécuter votre script, il suffit de taper la commande `>>> python nom-fichier.py` dans une fenêtre Terminal. Mais la plupart du temps, l'éditeur de texte de votre IDE possède une commande `run` qui exécute directement votre programme dans la fenêtre Terminal.

Exercice 1 : Premier pas

Vous pouvez ici travailler en mode interactif.

1-1) Assigner (affecter les deux termes sont identiques) les valeurs respectives 3, 5, 7 à trois variables `a`, `b`, `c`. Effectuez l'opération $a - b / c$. Interprétez le résultat obtenu.

1-2) Testez les lignes d'instructions suivantes. Décrire ce qui se passe :

```
>>>r, pi = 12, 3.14159
>>> s = pi * r **2
>>> print (s)
>>> print( type(r), type(pi), type(s) )
```

Quelle est, à votre avis, l'utilité de la *fonction* `type ()` ?

A présent écrivez vos programmes dans un script, penser à les sauvegarder !

1-3) On souhaite calculer l'augmentation de votre capital stocké à la banque. Si le taux d'intérêt annuel de votre banque vaut p , une quantité initiale d'argent A vaudra après n année :

$$A \left(1 + \frac{p}{100} \right)^n$$

Ecrire un programme pour calculer la somme d'argent disponible après 3 ans à partir d'une somme de 1000 euros et pour un taux d'intérêt annuel de 5 % (taux bien supérieur à la réalité...). Nommer votre programme `taux_interet.py`.

Note : Le module `Math` de python contient la plupart des fonctions mathématiques usuelles. Pour importer ce module, taper la commande `from math import*` en début de programme.

Exercice 2 : Instructions répétitives

2-1) Ecrivez un programme qui affiche les 20 premiers termes de la table de multiplication par 7, en signalant au passage (à l'aide d'une astérisque) ceux qui sont des multiples de 3. Penser à utiliser l'opérateur modulo %.

2-2) Ecrire un programme qui, a , b , c donnés, donne les solutions réelles (si elles existent) de l'équation algébrique bien connue: $ax^2 + bx + c = 0$.

2-3) Programmer récursivement le calcul du terme de rang n de la suite de Fibonacci définie par :

$$\begin{cases} u_0 = u_1 = 1 \\ u_{n+2} = u_{n+1} + u_n \end{cases}$$

Exercice 3 : Chaîne de caractères

3-1) On considère la chaîne de caractères suivante:

ch = " je crois pouvoir dire sans risque de me tromper que personne ne comprend la mécanique quantique. " (Citation du grand physicien Richard Feynman)

Ecrivez un script qui détermine si cette chaîne contient ou non le caractère « e ».

3-2) Ecrivez un script qui recopie une chaîne (dans une nouvelle variable) en insérant des astérisques entre les caractères. Ainsi, par exemple, « Darth Vader » deviendra « D*a*r*t*h* *V*a*d*o*r ».