

Algorithmie et programmation – feuille 1

Un langage de programmation comme Python est un code (un ensemble de règles et de commandes) qui est traduit en code binaire par un logiciel afin de pouvoir être utilisé par un ordinateur pour accomplir une ou plusieurs tâches.

Il existe beaucoup de programmes permettant de coder, en Python comme dans n'importe quel autre langage (C++, Java, etc.). En général, ils se composent d'un éditeur (une fenêtre dans laquelle on saisit le code), et d'un interpréteur (fenêtre affichant le résultat du code).

L'éditeur est la fenêtre dans laquelle on tape les commandes. L'interpréteur nous donne le résultat des commandes qu'on a entré, précédé des caractères `>>>`.

*La plupart des logiciels de programmation enregistrent les fichiers sous forme de fichier `py`, un standard de l'industrie. Le problème est que beaucoup le réinterprètent chaque fois qu'ils ouvrent un fichier, et un caractère spécial (accents, trémas, etc.) est souvent transcrit en code (par exemple, le caractère é donne `Ã©`). Pour cette raison, **il est préférable d'éviter tout caractère spécial** dans le codage comme dans le texte à afficher à l'écran.*

Un premier programme

Un programme exécute une ou plusieurs commandes au moyen d'un code. Connaître le vocabulaire et la grammaire de ce code est donc indispensable au codage. Si le vocabulaire varie avec la complexité d'un programme, la grammaire est toujours la même. Basée sur les mathématiques, elle est logique et rationnelle.

Par exemple, si on souhaite que le programme dise *Bonjour*, on entre dans l'éditeur la fonction **print()**:

Code	Résultat
<code>print("Bonjour")</code>	<code>>>></code> Bonjour

Une fonction est une commande qui dit à l'ordinateur quoi faire avec ce qui est contenu dans les parenthèses. Ce qu'on met entre parenthèses s'appelle l'argument de la fonction.

La fonction **print()** affiche à l'écran ce qui est contenu dans les parenthèses. Les guillemets indiquent à l'ordinateur d'ignorer le code contenu entre elles. Cela peut être utile quand on veut simplement écrire quelque chose.

Code	Résultat
<code>print(2*2)</code> <code>print("2*2")</code>	<code>>>></code> 4 2*2

On note que chaque ligne de code est interprétée l'une à la suite de l'autre. Un programme ne fait jamais deux choses à la fois, il exécute une série de commandes l'une après l'autre.

Sans guillemets, le programme tente d'interpréter ce qui est écrit. Si ce qui est écrit n'est pas du code (par exemple, « *bonjour* »), le programme ne comprendrait pas et un message d'erreur apparaîtrait.

On met entre guillemets ce qu'on veut que le programme n'interprète pas.

Afficher du texte à l'écran

En vous aidant de l'exemple ci-dessus, écrire une ligne de commande qui affiche votre nom et votre prénom.

Exemple :

```
>>>
Damien Devaux
```

Lignes non-interprêtées

Un programme étant souvent écrit par plusieurs programmeurs, il est important de faire passer des messages dans le code. Pour ce faire, on commence la ligne par le caractère croisillon #.

Code	Résultat
<pre>#Un calcul : print(2*2) #Un affichage écran print("Bonjour")</pre>	<pre>>>> 4 Bonjour</pre>

Le croisillon # permet d'entrer des messages qui sont ignorés par le programme et n'apparaissent pas dans l'interpréteur.

Insérer une ligne non-interprétée

Ajouter une ligne non-interprétée à votre code qui dit « Voici des calculs qu'on peut faire avec Python ».

Calculer en code

Un langage de programmation contient des fonctions et des instructions déjà définies. Nous avons déjà vu la fonction **print()**, qui permet d'afficher du texte à l'écran. Les opérateurs permettent de faire des opérations entre variables. Les principaux opérateurs mathématiques sont listés ci-contre.

On remarque que chaque ligne de code correspond à une ligne d'affichage. En effet, un langage de programmation exécute une commande à la fois, à chaque ligne. Si on veut que tout s'affiche sur une seule ligne, on peut combiner les arguments dans la fonction **print()**, séparés par des virgules.

Symbole		Exemple
+	addition	<pre>7+2 >>> 9</pre>
-	soustraction	<pre>7-2 >>> 5</pre>
*	multiplication	<pre>7*2 >>> 14</pre>
**	exposant	<pre>7**2 >>> 49</pre>
/	division	<pre>7/2 >>> 3,5</pre>
//	division entier	<pre>7//2 >>> 3</pre>
%	reste d'une division entière	<pre>7%2 >>> 1</pre>

Code	Résultat
<pre>print("2*2 est égal à",2*2)</pre>	<pre>>>> 2*2 est égal à 4</pre>

On peut utiliser la fonction print pour afficher du texte et des calculs sur une seule ligne en les séparant par des virgules.

Combiner un texte et du calcul sur une ligne.

En dessous du code que vous avez déjà entré, demander à l'interpréteur d'écrire sur une ligne un texte et un calcul, sans faire le calcul vous même. Utiliser l'exemple ci-dessus comme point de départ.

Exemple :

```
>>>
La somme de 3 et 4 est 7
```

Variables

En codage, il est utile et pratique d'utiliser des variables qui stockent de l'information et sont appelées dans des fonctions. Une variable est ainsi comme une boîte dans laquelle on stocke de l'information (un nombre, un texte, etc.).

On assigne une valeur à une variable à l'aide du signe égal. Lorsqu'on appelle une variable dans une fonction, celle-ci donne le contenu de la variable.

Code	Résultat
<pre>x = 3 print(x)</pre>	<pre>>>> 3</pre>

Une variable contient de l'information numérique ou textuelle. Les variables sont utilisées pour facilement changer une valeur dans un calcul ou dans un texte. On assigne une valeur à une variable à l'aide du signe égal.

Lorsqu'on nomme une variable, il est préférable d'éviter les accents et les caractères spéciaux.

Appeler une variable.

En dessous du code que vous avez déjà entré, définir une variable x comme ci-dessus et lui assigner une valeur (un nombre). Demander ensuite au programme de l'afficher dans une phrase complète.

Exemple :

```
>>>
```

La variable x est égale à 3.

Une variable peut être une lettre, mais aussi un mot, à condition d'éviter les accents et les caractères spéciaux (tout ce qui n'est pas l'alphabet simple ou les nombres entiers).

Code	Résultat
<pre>nombre = 3 print(nombre)</pre>	<pre>>>> 3</pre>

On peut aussi effectuer des calculs entre variables. Tout d'abord, on définit les variables qu'on va calculer entre elles (une variable par ligne). Ensuite, on instruit la fonction du calcul à opérer avec ces variables.

Code	Résultat
<pre>a=10 b=5 print(a,"puissance",b,"est égal à",a**b)</pre>	<pre>>>> 10 puissance 5 est égal à 100000</pre>

Note : vous pouvez utiliser l'exemple ci-dessus pour l'exercice suivant.

On peut aussi affecter des valeurs à plusieurs variables en une seule ligne à condition de, comme toujours, les séparer par des virgules.

Code	Résultat
<pre>x,y=2,3 print(x) print(y)</pre>	<pre>>>> 2 3</pre>

Maintenant, on peut effectuer plusieurs calculs entre variables d'un coup.

Définir et appeler plusieurs variables

Une variable peut être une lettre, mais aussi un mot, à condition d'éviter les accents et les caractères spéciaux.

Code	Résultat
<pre>nombre = 3 print(nombre)</pre>	<pre>>>> 3</pre>

On peut aussi effectuer des calculs entre variables. Tout d'abord, on définit les variables qu'on va calculer entre elles (une variable par ligne). Ensuite, on instruit la fonction du calcul à opérer avec ces variables.

Code	Résultat
<pre>a=10 b=5 print(a,"puissance",b,"est égal à",a**b)</pre>	<pre>>>> 10 puissance 5 est égal à 100000</pre>

Note : vous pouvez utiliser l'exemple ci-dessus pour l'exercice suivant.

On peut aussi affecter des valeurs à plusieurs variables en une seule ligne à condition de, comme toujours, les séparer par des virgules.

Code	Résultat
<pre>x,y=2,3 print(x) print(y)</pre>	<pre>>>> 2 3</pre>

Maintenant, on peut effectuer plusieurs calculs entre variables d'un coup.

Effectuer plusieurs calculs avec les mêmes variables.

En dessous du code que vous avez déjà entré, définir deux variables (par exemple, a et b) en leur attribuant deux valeurs numériques. En utilisant la fonction **print()**, afficher un texte qui décrit un calcul. Répéter 3 fois en exécutant un calcul différent à chaque fois.

Exemple :	<pre>>>> 10 puissance 5 est égal à 100000 10 divisé par 5 est égal à 2.0 10 moins 5 est égal à 5</pre>
-----------	-----------------------------------------------------------------------------------------------------------------