

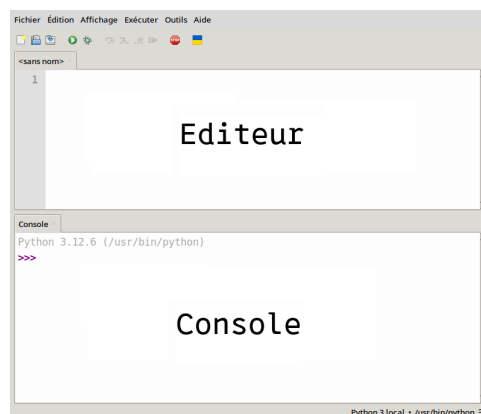
## I - TP 2

## Variables, conditions et boucle for

Python est un langage dit "interprété" : cela veut dire qu'un programme lit le code Python et se charge d'en exécuter les instructions. Nous allons utiliser **Thonny** pour écrire et exécuter les codes Python.

La fenêtre de Thonny est divisée en deux parties :

- l'**éditeur** dans lequel on écrira les codes python ;
- la **console** dans laquelle on peut exécuter les codes python.



1 On peut écrire des intructions dans la console. Pour exécuter une instruction, on appuie sur la touche Entrée.

1. Saisir dans la console les instructions ci-dessous.

|                     |                 |                           |               |
|---------------------|-----------------|---------------------------|---------------|
| • a = 2020          | • b = 0.1 + 0.2 | • type(c)                 | • 4 == 5      |
| • a                 | • type(b)       | • len(c)                  | • 4 == 4      |
| • type(a)           | • print(b)      | • c[2]                    | • 4 == "4"    |
| • print(a)          | • 7//3          | • c[0]                    | • 4 != 5      |
| • a = a + 4         | • type(7//3)    | • y = c + "tout le monde" | • 4 != 4      |
| • print("Promo", a) | • 7%3           | • print(y)                | • 4 < 5       |
| • a + 42            | • 7*2           | • d = 3.14                | • 4 < 4       |
| • print(a)          | • 7**2          | • type(d)                 | • 4 <= 4      |
| • 6 + 9             | • 7**3          | • e = 3,14                | • type(1 < 2) |
| • 7/3               | • c = "Bonjour" | • type(e)                 |               |
| • type(7/3)         | • c             |                           |               |

2. On suppose que x et y sont deux variables de type entier. u et v sont deux chaînes de caractères. Compléter le tableau ci-dessous.

| Instruction | Langage courant |
|-------------|-----------------|
| x = 3       |                 |
| type(x)     |                 |
| print(x)    |                 |
| x + y       |                 |
| x/y         |                 |
| x//y        |                 |
| x%y         |                 |

| Instruction                   | Langage courant |
|-------------------------------|-----------------|
| <code>x*y</code>              |                 |
| <code>x**y</code>             |                 |
| <code>x == y</code>           |                 |
| <code>x != y</code>           |                 |
| <code>x &lt;= y</code>        |                 |
| <code>(x, y)</code>           |                 |
| <code>len(u)</code>           |                 |
| <code>u[0], u[1], u[2]</code> |                 |
| <code>u + v</code>            |                 |

3. Calculer la valeur des expressions suivantes, puis vérifier à l'aide de la console.

- |                           |                                |                            |
|---------------------------|--------------------------------|----------------------------|
| a. <code>1.5 + 1.5</code> | e. <code>123 % 2</code>        | i. <code>len("snt")</code> |
| b. <code>1,5 + 1,5</code> | f. <code>3.0*7</code>          | j. <code>"snt"[0]</code>   |
| c. <code>5/4</code>       | g. <code>4 - (3 - 3**2)</code> | k. <code>"a" + "b"</code>  |
| d. <code>5//4</code>      | h. <code>2**5</code>           | l. <code>"a" * 3</code>    |

4. Calculer la valeur booléenne des expressions suivantes (True ou False), puis vérifier à l'aide de la console.

- |                             |                           |                              |                          |
|-----------------------------|---------------------------|------------------------------|--------------------------|
| a. <code>3*3 &lt; 10</code> | b. <code>3*7 == 21</code> | c. <code>5**2-5 != 20</code> | d. <code>not True</code> |
|-----------------------------|---------------------------|------------------------------|--------------------------|



**2** Plutôt que d'écrire les instructions les unes après les autres dans la console, on les écrit toutes dans un fichier (appelé code source) à l'aide de l'**éditeur**. Une fois que toutes les instructions ont été écrites, on peut exécuter le programme.

1. Recopier dans l'éditeur toutes les instructions ci-dessous.

Code python

```

1  2*5
2  print(3*5)
3  a = 4*5
4  a = 3*4
5  print(a)
```

2. a. Exécuter le programme. Appuyer pour cela sur la flèche verte ou bien la touche F5.  
b. Que se passe-t-il ? La console affiche-t-elle le résultat de tous les calculs ? Pourquoi ?
3. a. Entrer en mode débogueur :
- Dans le menu "Affichage", cliquer sur l'item "Variables".
  - Démarrer le mode débogueur en cliquant sur l'icône en forme d'insecte .
  - Exécuter pas à pas le programme en cliquant sur l'icône .
- b. Pourquoi est-il plus facile de trouver des bugs à l'aide du mode débogueur ?

3

1. Créer un nouveau fichier correspondant à cet exercice.

2. a. Recopier et exécuter le code suivant.

```
Code python
1 matiere = "anglais"
2 if matiere == "NSI":
3     print("Trop cool")
4     print("J'adore")
```

Commenter.

b. Modifier **uniquement la ligne 1 et l'indentation** du code de telle sorte que l'affichage produit soit :
 Résultat

```
Trop cool
J'adore
```

3. a. Recopier et exécuter le code suivant.

```
Code python
1 x = -4
2 if x > 0:
3     print("positif")
4 else:
5     print("négatif ou nul")
6     print("fin")
```

b. Modifier uniquement la première ligne du code afin que le mot positif soit affiché par votre programme.

4. Il est possible de préciser plusieurs alternatives, à l'aide du mot-clé elif (contraction de else et de if).

```
Code python
1 t = 40
2 if t > 100:
3     print("évaporation")
4 elif t > 37.5:
5     print("très chaud")
6 elif t > 20:
7     print("agréable")
8 else:
9     print("froid")
```

```
Code python
1 t = 40
2 if t > 20:
3     print("agréable")
4 elif t > 37.5:
5     print("très chaud")
6 elif t > 100:
7     print("évaporation")
8 else:
9     print("froid")
```

Recopier et exécuter les codes ci-dessus. Commenter l'affichage réalisé par les deux programmes.

4

1. Recopier et exécuter chacun des codes ci-dessous.

```
Code python
1 for i in range(3):
2     print("Bonjour")
```

```
Code python
1 for i in range(5, 30, 5):
2     print(i)
```

```
Code python
1 for i in range(3):
2     print(i)
```

```
Code python
1 for i in range(3):
2     for j in range(3):
3         print(j)
```

```
Code python
1 for i in range(2, 6):
2     print(i)
```

```
Code python
1 for i in range(3):
2     for j in range(3):
3         print(i)
```

2. Décrire le plus précisément possible l'effet des lignes :

a. `for i in range(n):`

b. `for i in range(m, n):`

c. `for i in range(m, n, p):`

d. `for i in range(m, n, p):`

avec  $m > n$  et  $p < 0$

3. a. Recopier puis exécuter le code suivant.

Code python

```
1 c = ""
2 for j in range(3):
3     c = c + "*"
4 print(c)
```

b. Recopier puis exécuter le code suivant.

Code python

```
1 for i in range(1, 4):
2     c = ""
3     for j in range(i):
4         c = c + "*"
5     print(c)
```

c. Écrire un code python permettant d'afficher le dessin suivant.

 Résultat

```
|---|
|-----|
|-----|
```

5

1. Recopier et exécuter le code ci-dessous.

Code python

```
1 c = ""
2 for j in range(5):
3     if j < 3:
4         c = c + "_"
5     else:
6         c = c + "*"
7 print(c)
```

2. Écrire les codes python permettant d'afficher les dessins suivants. Il faut pour cela utiliser le mot-clé `for` et parfois le mot-clé `if`.

 Résultat

```
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
*****
```

 Résultat

```
* _
** _
*** _
**** _
***** _
***** _
***** _
***** _
***** _
***** _
***** _
```

 Résultat

```
-----
-----*
-----**
-----***
-----****
-----*****
-----*****
-----*****
-----*****
-----*****
-----*****
```

 Résultat

```
*****
* _ *
* _ *
* _ *
* _ *
* _ *
* _ *
* _ *
* _ *
* _ *
* _ *
*****
```

 Résultat

```
*****
** _ *
* _ *
* _ *
* _ *
* _ *
* _ *
* _ *
* _ *
* _ *
*****
```