

# Algo Python - Exercices type - Variables, boucles for

Remarque : Lorsque ce n'est pas précisé, le langage de utilisé est Python.

## I INTRODUCTION - SYNTAXE DE BASE

EXERCICE I — Écrire un algorithme qui affiche précisément *Hello World!*

EXERCICE II — Corrigez les erreurs dans le programme ci-dessous, afin qu'il affiche Bonjour.

```
print(Bonjour)
```

EXERCICE III — Les deux algorithmes suivant sont-ils identiques ? `print("Bonjour")` | `print("Bonjour!")`

EXERCICE IV — Écrire un programme qui affiche exactement le texte qui suit :

```
Bonjour
Je m'appelle Hélène
Je suis une fille
Comme les autres
```

EXERCICE V — (\*, un peu délicat) Les deux programmes suivant vont-ils afficher la même chose ?

Algo 1	Algo 2
<pre>print("Un") print("Deux") print("Trois") print("Soleil")</pre>	<pre>print("Un ", end = "") print("Deux ", end = "") print("Trois ", end = "") print() print("Soleil")</pre>

## II VARIABLE

EXERCICE VI — Que fais l'algorithme suivant ?

```
monAge = 25
print("J'ai", monAge, " !")
```

EXERCICE VII — La distance entre la Terre et la Lune est d'environ 55 758 000 km. Que fait l'algorithme suivant ?

```
distance = 55758000
print(distance)
print(2*distance)
```

EXERCICE VIII — (important)

1. Parmi les noms de variables suivants, lesquels sont autorisés, et lesquels sont interdits ?  
maVariable ; distanceMaisonLycee ; 123Soleil ; Soleil123 ; J'ai\_faim, J\_ai\_faim ; Trop!Bien
2. maVariable, MaVariable et mavariable représente-t-elle la même variable ?

EXERCICE IX — (important) Qu'affiche le programme suivant ?

```
contenance = 60
print("Départ :", contenance)
contenance = 100
print("Remplissage :", contenance)
contenance = contenance - 15
print("Consommation :", contenance)
```

EXERCICE X — (important) Une cour de récréation en forme de carrée a été mesurée avec quatre bâtons de longueurs respectives 17m, 7m et 2m. La longueur du côté de la cour est égale à 5 fois le premier bâton plus 2 fois le second plus 1 fois le troisième plus 2 fois le quatrième.

Écrire un programme qui affiche deux lignes : la première doit contenir la surface de la cour (en  $m^2$ ), et la seconde ligne doit contenir son périmètre (en mètre).

EXERCICE XI — (extrêmement important) Parmi les programmes suivants, lesquels sont valides ? Pour les non valides, dire ce qui ne va pas ; pour les programmes valides, préciser ce qu'affichent les algorithmes valides.

```
Algo 1
nombreDeFrere = 2
nombreDeSoeur = 0
print(nombreDeFrere + nombreDeSoeur)
```

```
Algo 2
nombreDeFrere = 2
print(nombreDeSoeur)
```

```
Algo 3
monAge = 15
monAge = 16
print(monAge)
```

```
Algo 4
age = 15
age = age+1
print(age)
```

```
Algo 5
1 = 1
print(1)
```

```
Algo 6
monAge = 15
monAge-1 = 15
print(monAge)
```

```
Algo 7
monAge = tonAge + 1
tonAge = 15
print(monAge)
```

### III ITÉRATION - BOUCLE FOR

EXERCICE XII — (basique) Écrire un programme qui affiche 100 fois la phrase *Je dois être sage en cours de maths et écouter le prof.*

EXERCICE XIII — (basique) Écrire un programme qui affiche 32 fois *Bonjour* et une fois *Au revoir!*

EXERCICE XIV — (important) Corrigez les erreurs contenues dans le programme ci-dessous afin qu'il affiche 13 fois de suite le texte  $9*8=72$ .

```
for loop in range(13)
print("9 * 8 = 72")
```

EXERCICE XV — (important) Écrire un programme qui affiche tous les nombres entiers de 1 à 100, puis qui affiche la phrase *Ouf! Fini, mais c'était long.*

**Correction** On propose deux solutions.

```
compteur = 0
for loop in range(100) :
    compteur = compteur + 1
    print(compteur)
print("Ouf! Fini, mais c'était long")
```

Dans cette algorithmes, on a une variable appelée *compteur*, qui compte. Dans la boucle, ajoute 1 à chaque étape au compteur, puis on affiche le compteur. Comme on répète 100 fois la boucle, et que l'on commence à compter à 1, on a bien tous les nombres de 1 à 100.

Voici une deuxième possibilité :

```
for loop in range(100) :
    print(loop+1)
print("Ouf! Fini, mais c'était long")
```

Ici, on utilise le fait que dans l'instruction `for loop in range(100)`, `loop` est en fait une variable qui va prendre toutes les valeurs entières entre 0 et 99. Pour pouvoir compter de 1 à 100, il suffit donc d'afficher `loop+1` (`loop+1` car `loop` part de 0 et arrive à 100, alors que l'on veut compter de 1 à 100). C'est un point délicat de Python, et sujet à beaucoup d'erreurs bêtes : on commence à compter de 0! □

EXERCICE XVI — (\*, un peu plus délicat pour l'instant) Écrire un programme qui compte à l'envers Votre programme devra afficher tous les nombres de 100 jusqu'à 0. Ensuite, il affichera *Décollage!*

EXERCICE XVII — (\* plus dur, mais important) Ecrire un programme qui affiche :

```
0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9
```