



Les boucles en PYTHON



Olivier CHAUMETTE
professeur de physique-chimie et d'informatique
Lycée Jean-Paul Sartre – 69 BRON
olivier.chaumette@ac-lyon.fr

Notion de boucle bornée

Pour exécuter plusieurs fois une instruction, on utilise une **BOUCLE**

EXEMPLE:

Dans l'éditeur, taper les instructions suivantes et exécuter:

```
for i in range (0,4):  
    print ("Bonjour")  
print ("Au revoir")
```

Ne pas oublier les deux points

Ne pas oublier d'indenter cette ligne

Que se passerait-il si la dernière ligne était elle aussi indentée ? Essayer

EXEMPLE:

Dans l'éditeur, taper les instructions suivantes et exécuter:

```
for i in range (0,4):  
    print ("Bonjour", i)
```

Bien comprendre à quoi correspondent 0 et 4 dans le "in range"



Modifier le 0 et le 4 pour comprendre

Bornes d'une boucle bornée

Le compteur de boucle **i** prends des valeurs entre un début et une fin. On dit que la boucle est **bornée** (on sait comment de fois elle va se faire) ou bien "**boucle if**"

```
for i in range(début, fin) :
```

La variable nombre **i** va prendre des valeurs entre **début** et **fin-1**.

La boucle va donc se faire **fin-début** fois

EXEMPLE:

Dans l'éditeur, taper un programme qui affiche:

```
>> élève n°1  
>> élève n°2  
>> élève n°3
```

etc... jusqu'à:

```
>> élève n°35
```

EXERCICE

- Créer un nouveau module (menu Fichier >> Nouveau >> Nouveau Module Python)
- Sauvegarder le module dans votre dossier personnel dans le dossier SNT sous le nom:
Exercice_python3

Dans l'éditeur, écrire un programme qui:

1. *Demande un nombre (et le met dans une variable de type nombre appelée "n")*
2. *Affiche "Le nombre n vaut:" suivi de la valeur de n*
3. *Crée un boucle tournant 5 fois au cours de laquelle:*
 - *n prend la valeur n+2*
 - *affiche "maintenant, n vaut" suivi de la valeur de n*
4. *Affiche, une fois la boucle terminée, "La boucle est bouclée ! Et n vaut" suivi de la valeur de n*



**APPELER LE PROFESSEUR POUR LUI
MONTRER VOTRE PROGRAMME**

EXERCICE 2

- Créer un nouveau module (menu Fichier >> Nouveau >> Nouveau Module Python)
- Sauvegarder le module dans votre dossier personnel dans le dossier SNT sous le nom: Exercice_python4

Dans l'éditeur, écrire un programme qui affiche la table de multiplication d'un chiffre :

il demande un chiffre à l'utilisateur et si c'est « 3 » par exemple, il affiche « la table de multiplication de 3 est : » puis en dessous les 10 multiples de 3.



**APPELER LE PROFESSEUR POUR LUI
MONTRER VOTRE PROGRAMME**

pour les plus rapides

*Améliorer le programme en affichant,
par exemple : $3 \times 1 = 3$, $3 \times 2 = 6$ etc...*

Notion de boucle non bornée

Parfois, on doit exécuter plusieurs fois des instructions sans savoir à l'avance combien de fois il faudra le faire. On dit alors que la boucle est **non bornée** (on l'appelle souvent "**boucle while**")

Exemple (en français):

```
Je regarde une première fois la météo
Tant qu'il fait beau:
    je joue dehors
    je regarde le ciel
Je rentre car il pleut
```

Boucle non bornée qui "tourne" tant qu'il fait beau.

Elle s'arrêtera dès qu'il fera mauvais

Dans cet exemple simple, il est indispensable de regarder le ciel au sein de la boucle sinon elle ne s'arrêtera jamais (puisque l'on ne saura jamais qu'il ne fait plus beau)

EXERCICE

Dans un nouveau module, taper un programme qui:

- *Demande un nombre à l'utilisateur et le met dans une variable de type nombre appelée n.*
- *Crée une boucle "tant que" n est différent de 10.*
- *Dans la boucle demande un nombre à l'utilisateur et le met dans la variable n*
- *En dehors de la boucle, affiche "Vous avez saisi 10"*

Essayer le programme et comprendre pourquoi la boucle s'arrête



APPELER LE PROFESSEUR POUR LUI MONTRER VOTRE PROGRAMME ET LUI DIRE POURQUOI LA BOUCLE S'ARRÊTE

EXERCICE POUR LES PLUS RAPIDES

Dans un nouveau module, taper le code suivant:

```
import random
de = random.randint(1, 6)
print(de)
nb_lancer = 1
```

Ligne du
tirage au sort

Ce code tire un nombre au hasard entre 1 et 6 et met ce nombre dans la variable "de"

A la suite, taper un programme qui:

- *Crée un boucle tant que la valeur du dé ne vaut pas 6*
- *Dans cette boucle:*
 - * *on retire au sort entre 1 et 6 et on met la valeur dans "de"*
 - * *nb_lancer est augmenté de 1*
 - * *On affiche n*
- *A la fin de la boucle, on affiche "il a fallu ... lancers pour obtenir un 6"*



**APPELER LE PROFESSEUR POUR LUI
MONTRER VOTRE PROGRAMME ET LUI
EXPLIQUER CE QU'IL FAIT**

POUR ALLER PLUS LOIN, POUR LES PLUS RAPIDES

Il est possible de faire varier un compteur de boucle bornée de 2 en 2 (par exemple).

Exemple: `for i in range(0,10,2)`

"2" s'appelle
le pas
(d'itération)

`for i in range(début, fin, pas) :`

La variable nombre *i* va prendre des valeurs entre *début* et *fin-1* par "sauts" de *pas*.

Il est aussi possible de faire diminuer un compteur de boucle en mettant un pas de -1

Exemple: `for i in range(10,0,-1)`

Attention: la 1^{ère} borne doit être supérieure à la seconde

EXERCICES POUR LES PLUS RAPIDES

Dans un nouveau module, taper un programme qui:

- 1. Ecrire un programme qui affiche « Compter de 3 en 3 c'est facile ! » et qui affiche dessous :3 6 9 etc... jusqu'à 30*
- 2. Ecrire un programme qui affiche « Trop facile de compter de 2 en 2 à l'envers ! » et qui affiche dessous :20 18 etc*



**APPELER LE PROFESSEUR POUR LUI
MONTRER VOTRE PROGRAMME**

EXERCICES POUR LES TRÈS RAPIDES

*Dans un nouveau module, taper un programme qui affiche les dessins d'**ASCII art** suivants, en utilisant des boucles:*

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

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

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

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

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

AIDE:

Pour afficher, par exemple, 4 fois la lettre "A", il suffit de taper

```
print (4*"A")
```



APPELER LE PROFESSEUR POUR LUI
MONTRER VOTRE PROGRAMME

EXERCICES POUR LES TRÈS TRÈS RAPIDES

JEU du "plus grand ou plus petit"

Dans un nouveau module, créer un programme qui:

- *Tire un nombre au hasard entre 1 et 100 (et le met dans "n")*
- *Demande un nombre à l'utilisateur*
- *Tant que le nombre entré est différent du nombre tiré au sort:
On dit si le nombre est plus grand ou plus petit que n
(utiliser les conditions)
On redemande un nombre à l'utilisateur.*

A la fin, on affiche "gagné, le nombre était bien:" suivi du nombre n



**APPELER LE PROFESSEUR POUR LUI
MONTRER VOTRE PROGRAMME**