

# Les boucles non bornées

Site: ISN Space

Cours: Les boucles en python

Livre: Les boucles non bornées

Imprimé par: Kazi-Aoual Chawky

Date: Saturday 2 May 2020, 12:34

# Table des matières

1. La boucle While
2. Comment choisir entre for et while ?
3. Les mots-clefs break, continue et else

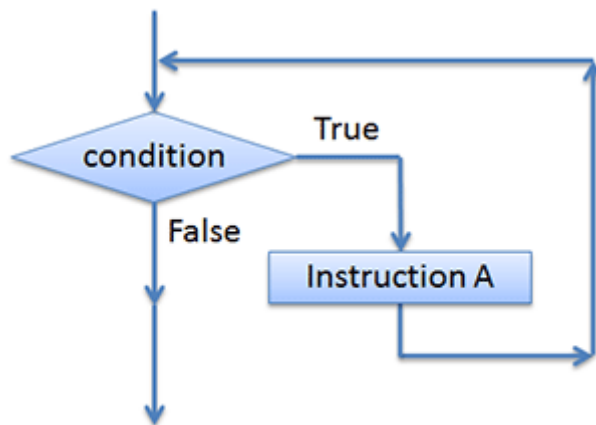
# 1. La boucle While

## Boucle while

Si on ne connaît pas à l'avance le nombre de répétitions, on choisit une boucle non bornée **while** .

**Syntaxe :**

```
while condition:  
    Instruction A
```



**Exemple d'utilisation :**

```
x = 1  
while x < 10:  
    print("x a pour valeur", x)  
    x = x * 2  
print("Fin")
```

**Tester l'exemple**

**Affichage après exécution :**

```
x a pour valeur 1  
  
x a pour valeur 2  
x a pour valeur 4  
x a pour valeur 8  
Fin
```

Le mot-clé **while** se traduit par **tant que** en français. Le corps de la boucle (c'est-à-dire le bloc d'instructions indentées) sera répété tant que la **condition** est vraie.

Dans l'exemple ci-dessus, **x** sera multiplié par 2 tant que sa valeur reste inférieure à 10.



Si la condition est fausse au départ, le corps de la boucle n'est jamais exécuté. Si la condition reste toujours vraie, alors le corps de la boucle est répété indéfiniment.

### À toi de jouer :

Écrire un script qui permet d'afficher les carrés des nombres de 1 à 9.

Le résultat devrait ressembler à ceci :

```
1
4
9
16
25
36
49
64
81
```



#### **Attention !**

N'oubliez pas d'incrémenter le compteur, sinon la boucle continuera indéfiniment. Dans le premier exemple, la variable **x** est le compteur et a été initialisé à 1.

**Tester votre programme**

## 2. Comment choisir entre for et while ?

### Comment choisir entre boucle for et boucle while ?

En général, si on connaît avant de démarrer la boucle le nombre d'itérations à exécuter, on choisit une boucle **for** . Au contraire, si la décision d'arrêter la boucle ne peut se faire que par un test, on choisit une boucle **while** .



Il est toujours possible de remplacer une boucle for par une boucle while.

### Transformation d'une boucle for en un boucle while

```
for i in range(4):  
    print("i a pour valeur", i)
```

Le programme ci-dessus est équivalent à:

```
i = 0  
while i < 4:  
    print("i a pour valeur", i)  
    i = i + 1
```

**Tester le programme**

# 3. Les mots-clefs break, continue et else

## L'instruction break

Avec l'instruction **break** , nous pouvons arrêter la boucle même si la condition **while** est vraie :

**Exemple :**

Quittez la boucle quand j'ai 6 ans :

```
i = 1
while i < 12:
    print(i)
    if i == 6:
        break
    i += 1
```

**Tester l'exemple**

## L'instruction continue

Avec l'instruction **continue** , nous pouvons arrêter l'itération en cours et continuer avec la suivante :

**Exemple :**

Passez à l'itération suivante si j'ai 6 ans :

```
i = 0
while i < 12:
    i += 1
    if i == 6:
        continue
    print(i)
```

**Tester l'exemple**

## L'instruction else

Avec l'instruction **else** , nous pouvons exécuter un bloc de code une fois lorsque la condition n'est plus vraie:

**Exemple :**

Imprimez un message une fois que la condition est fausse:

```
i = 1
while i < 12:
    print(i)
    i += 1
else:
    print("i est plus grand que 12")
```

**Tester l'exemple**