

# Boucles (for, for each, while, do...while)

Pour ne pas avoir besoin de repeter un code beaucoup de fois, on peut utiliser des boucles. Il en existe 4 différentes.

- `while` qui exécute un code en boucle tant qu'une certaine condition est remplie
- `do...while` qui exécute une fois le code, puis la répète encore tant que la condition est remplie
- `for` qui permet d'exécuter un code un nombre définis de fois
- `for each` qui permet d'exécuter un code pour chaque élément dans une collection (par exemple un tableau comme vu dans un chapitre précédent)

## La boucle while

```
// Compter jusqu'a 42
int i = 0;
while (i <= 42) {
    System.out.printf("Le compteur i est à %d.\n", i);
    i++;
}

// Ne va rien faire car la condition n'est pas remplie
int j = 0;
while (i == 666) {
    System.out.printf("Le compteur j est à %d.\n", j);
    j++;
}
```

Le premier bloc de code va s'exécuter et on verra 43 lignes allant de 0 à 42. Le deuxième bloc de code ne s'exécutera pas car sa condition que i doit être égal à 666 ne sera pas remplie.

## La boucle do...while

```
// Compter jusqu'a 42
do {
    System.out.printf("Le compteur i est à %d.\n", i);
    i++;
} while (i <= 42);

// Ne va faire qu'une seule itération car la condition ne sera pas remplie
int j = 0;
do {
    System.out.printf("Le compteur j est à %d.\n", j);
    j++;
} while (j == 666);
```

Dans ce cas ci, comme avant, le premier bloc de code s'exécutera 43 fois de 0 à 42. Tandis que le second bloc de code ne s'exécutera qu'une fois et indiquera "0".

Contrairement à while, la boucle do...while s'exécutera toujours un minimum de une fois. Puis se réexécutera si/tant que la condition est remplie.

## La boucle for

```
// Compter jusqu'a 42
for (int i = 0; i <= 42; i++) {
    System.out.printf("Le compteur est à %d.\n", i);
}
```

Ce code indiquera 43 lignes allant de 0 à 42. C'est comme une version contractée de la boucle while que nous avons déjà vu plus tot.

La différence est que le nombre d'itération est déjà prédéfinis. On définit une boucle for comme ceci :

```
for (initialisation; condition; modification) {
    // code à exécuter dans la boucle
}
```

## La boucle for each

```
// Lire chaque élément d'un tableau
String[] tableau = { "un", "deux", "trois", "quatre", "cinq", "six", "sept", "huit", "neuf", "dix" };

for (String element: tableau) {
    System.out.println(element);
}
```

La boucle for each permet de passer en revue tous les éléments d'une *collection*, par exemple dans ce cas ci, d'un tableau (voir dans un chapitre précédent). Dans ce code on assigne pour chaque itération la variable "element" qui correspond à l'élément en cours dans la variable "tableau".

## En savoir plus

- [Wikiversity FR - Boucles et structures conditionnelles](#)

---

Revision #1

Created 27 April 2023 06:31:31 by SnowCode

Updated 27 April 2023 06:35:12 by SnowCode