

Les Boucles

Le rôle des boucles est de répéter un certain nombre de fois les mêmes opérations. Tous les programmes, ou presque, ont besoin de ce type de fonctionnalité. Une boucle s'exécute tant qu'une condition est remplie.

La boucle while

Syntaxe

```
while (/* Condition */)
{
    //Instructions à répéter
}
```

Exemple

```
String prenom;
char reponse = 'O';
Scanner sc = new Scanner(System.in);
while (reponse == 'O')
{
    System.out.println("Donnez un prénom : ");
    //On récupère le prénom saisi
    prenom = sc.nextLine();
    //On affiche notre phrase avec le prénom
    System.out.println("Bonjour " + prenom + ", comment vas-tu ?");
    //On demande si la personne veut faire un autre essai
    System.out.println("Voulez-vous réessayer ? (O/N)");
    //On récupère la réponse de l'utilisateur
    reponse = sc.nextLine().charAt(0);
} //Fin de la boucle

System.out.println("Au revoir...");
```

La boucle do... while

Syntaxe

```
do{
    //Instructions
}while(a < b);
```

Différence

- La boucle do... while s'exécutera *au moins une fois*, contrairement à while.
- Une différence de syntaxe, qui se situe après la condition du while. Il y a un « ; » après le while. C'est

```
String prenom = new String();
char reponse = ' ';

Scanner sc = new Scanner(System.in);

do{
    System.out.println("Donnez un prénom : ");
    prenom = sc.nextLine();
    System.out.println("Bonjour " + prenom + ", comment vas-tu ?");

    do{
        System.out.println("Voulez-vous réessayer ? (O/N)");
        reponse = sc.nextLine().charAt(0);
    } while(reponse != 'O' && reponse != 'N');

}while (reponse == 'O');

System.out.println("Au revoir...");
```

La boucle for

Syntaxe

```
for(int i = 1; i <= 10; i++)
{
    System.out.println("Voici la ligne "+i);
}
```

```
for(int i = 10; i >= 0; i--)
    System.out.println("Il reste "+i+" ligne(s) à écrire");
    Boucle for avec décrémentation
```

En résumé

- Les boucles permettent simplement d'effectuer des tâches répétitives.
- Il existe plusieurs sortes de boucles :
 - La boucle **while(condition){...}** évalue la condition puis exécute éventuellement un tour de boucle (ou plus) ;
 - La boucle **do{...} while(condition);** fonctionne exactement comme la précédente, mais effectue un tour de boucle quoi qu'il arrive ;
 - La boucle **for** permet d'initialiser un compteur, une condition et un incrément dans sa déclaration afin de répéter une partie de code un nombre limité de fois.
- Tout comme les conditions, si une boucle contient plus d'une ligne de code à exécuter, il faut en délimiter le début et la fin.