

# JAVASCRIPT: LES BOUCLES 1

En JavaScript, il est possible de faire répéter à l'ordinateur une série d'instructions tant qu'une condition est respectée. Nous appelons cela des **boucles**.

Il existe plusieurs sortes dont les plus utilisées sont :

- while
- for

Nous reviendrons sur d'autres boucles plus tard.

## WHILE

Exemple 1 : obliger l'utilisateur à saisir un nombre

```
let age = window.prompt("Quel est votre âge ?");  
while (isNaN(age)) {  
  age = window.prompt("Merci de répondre un nombre. Quel est votre âge ?");  
}  
console.log("Vous avez " + age + " ans.");
```

Ici, nous demandons l'âge de l'internaute. Puis nous disons :

« Tans que l'âge n'est pas un nombre (isNaN) , alors on demande l'âge à nouveau »

La boucle ne va s'arrêter que une fois que la condition, dans la parenthèse du while, n'est plus respectée, c'est-à-dire dès que l'âge sera un nombre.

Exemple 2 : exécuter la boucle un nombre défini de fois

```
let cpt = 0;  
while (cpt<=20) {  
  cpt++;  
  console.log(cpt);  
}
```

Ici la boucle va s'exécuter tant que le compteur est plus petit que 20. Entre chaque tour de boucle, on ajoute 1 au compteur avec cpt++.

Attention à ne pas oublier cette incrémentation, car si la valeur de cpt ne change pas, alors vous aurez une boucle infinie, qui ne se termine jamais parce que cpt est toujours inférieur à 20.

## FOR

Cette boucle est un peu particulière, elle a dans ses paramètres un index, une condition et une incrémentation. C'est la boucle qu'on utilise pour parcourir un tableau bien souvent.

**Syntaxe :**

```
for(let i=0 ; i<liste.length ; i++){  
    console.log(liste[i]);  
}
```

*Initialisation* → let i=0 → initialisation de l'index qui va être incrémenté

*Condition* → i<liste.length → cette condition qui doit être respectée pour que la boucle continue

*Incrémentation* → i++ → syntaxe d'incrément de notre index

Cette boucle va passer dans chaque case de notre tableau liste et afficher ce qu'elle contient en console.

## PORTEE DE LA VARIABLE OU SCOPE

Dans l'exemple précédent, nous déclarons la variable i dans la boucle « for ». Les variables déclarées avec let ne sont utilisables que dans la structure dans laquelle elles sont déclarées.

Si je fais ça par exemple :

```
for(let i=0 ; i<liste.length ; i++){  
    console.log(liste[i]);  
}  
  
console.log(i)
```

Voilà ce que me répond la console :

❗ ▶ ReferenceError: i is not defined

Elle ne connaît pas « i » puisque cette dernière a été déclarée dans la boucle for et n'est donc utilisable qu'au sein de cette boucle.

C'est valable ici pour les boucles for mais aussi pour toute autre bloc, les if, les while....