

Javascript_Part3: Les boucles

Les boucles sont des structures qui permettent de répéter une séquence du programme un nombre de fois spécifique qui peut être soit défini, soit lié à une condition. Effectuer une boucle en javascript peut se faire avec les instructions for, while et do while.

Les instructions "break" et "continue" permettent de gérer les passages et les sorties des boucles :

- break : provoque la sortie immédiate de la boucle courante
- continue : interrompt le code restant pour la valeur en cours et provoque le passage à la valeur suivante

La boucle for

Syntaxe de la boucle for :

```
for(compteur initial ; condition compteur ; increment compteur )
{
  bloc de code
}
```

La structure for permet d'exécuter le bloc de code autant de fois que la condition remplie par le compteur est respectée.

- compteur initial : valeur initiale du compteur
- condition compteur : condition impliquant l'exécution du code dans la boucle lorsqu'elle est remplie
- increment : valeur pouvant être positive ou négative appliquée au compteur à chaque fin de passage dans la boucle

Exemples d'utilisation de la boucle for :

```
for (i = 0; i < 10; i++)
{
  document.write(i + ' » ');
}
```

```
for (i = 10; i > 0; i--)
{
  document.write(i + ' « ');
}
document.write("0");
```

Résultat de l'exemple :

0 » 1 » 2 » 3 » 4 » 5 » 6 » 7 » 8 » 9 » 10 « 9 « 8 « 7 « 6 « 5 « 4 « 3 « 2 « 1 « 0

```
// Déclaration d'un tableau contenant les jours de la semaine
var arrJour = new Array("Lundi", "Mardi", "Mercredi", "Jeudi", "Vendredi", "Samedi", "Dimanche");
for (i = 0; i < arrJour.length; i++){
  document.write("<br />" + i + " - " + arrJour[i]);
}
```

Résultat de l'exemple :

```
0 - Lundi
1 - Mardi
2 - Mercredi
3 - Jeudi
4 - Vendredi
5 - Samedi
6 - Dimanche
```

La boucle for peut également être employée avec le mot clé in pour parcourir les éléments d'une énumération ou d'un objet par exemple.

Exemples d'utilisation de la boucle for / in :

```
// Déclaration d'un tableau contenant les jours de la semaine
var arrJour = new Array("Lundi", "Mardi", "Mercredi", "Jeudi", "Vendredi", "Samedi",
"Dimanche");
for (val in arrJour){
    document.write("<br />" + val + " - " + arrJour[val]);
}
```

Résultat de l'exemple :

```
0 - Lundi
1 - Mardi
2 - Mercredi
3 - Jeudi
4 - Vendredi
5 - Samedi
6 - Dimanche
```

La boucle while

Syntaxe de la boucle while :

```
while(condition)
{
    bloc de code
}
```

La boucle while permet d'exécuter en chaîne un bloc de code tant que la condition spécifiée est vérifiée. il faut faire attention car la variable utilisée pour effectuer la comparaison doit évoluer au sein du code de la boucle sans quoi celle-ci sera infinie.

Très fréquemment la comparaison est faite sur une valeur numérique et il faut dans ce cas à la fin de la boucle incrémenter cette valeur.

Exemple d'utilisation de la boucle while :

```
var i = 0;
while(i < 5)
{
    document.write("<br />i vaut " + i);
    i++;
}
```

Résultat de l'exemple :

```
i vaut : 0
```

```
i vaut : 1  
i vaut : 2  
i vaut : 3  
i vaut : 4
```

La boucle do / while

Syntaxe de la boucle do / while :

```
do  
{  
  bloc de code  
}  
while(condition);
```

Le bloc de code de l'instruction do / while sera au moins exécuté une fois, même si la condition décrite au niveau du while n'est pas respectée, car cette condition n'est testée qu'après un premier passage dans le code (Contrairement à la boucle while qui effectue le test avant de rentrer dans le bloc de code).

Exemple d'utilisation de la boucle do / while :

```
var i = 8;  
do  
{  
  document.write("<br />i vaut " + i);  
  i++;  
}  
while(i < 5)
```

Résultat de l'exemple :

```
i vaut : 8
```