

Développement d'applications Web

Les Boucles en PHP

Les Boucles en PHP [1]

Il existe quatre types de boucles en PHP :

- `while` ,
- `do ... while` ,
- `for`
- `foreach` .

Nous allons voir chacune de ces boucles avec à chaque fois les deux mêmes exemples : afficher des chiffres de 0 à 5 inclus ; afficher les valeurs d'un tableau.

Boucle while en PHP

La boucle while est l'une des deux les moins utilisées. Elle permet, comme son nom l'indique, d'exécuter une action « tant que » une condition est vraie.

Par exemple, on peut prendre une variable `$i` initialisée à 0. La condition sera « tant que la variable est inférieure à 6 », alors on exécute une action « afficher la valeur de `$i` et l'incrémenter. »

Au terme de syntaxe, l'écriture est la suivante : `while(condition) { action }`.

Exemple 1

```
$i = 0;

while( $i < 6 ) {

    echo "- {$i}<br />";

    $i++;

}

/**
```

- 0

- 1

- 2

- 3

- 4

- 5

*/

Exemple 2

```
$list = [ 'C', 'o', 'd', 'e', 'u', 'r' ];
```

```
$i = 0;
```

```
while( $i < 6 ) {
```

```
    echo "{$list[$i]}.";
```

```
    $i++;
```

```
}
```

```
/** C.o.d.e.u.r. */
```

Boucle do ... while en PHP

Tout comme la boucle while, do ... while est très peu utilisée. Son fonctionnement est extêment le même ! La seule chose qui chose, c'est la syntaxe. On va d'abord commencer par renseigner l'action puis donner la condition : do { action } while(condition);.

Exemple 1

```
$i = 0;
```

```
do {
```

```
    echo "- {$i}<br />";
```

```
    $i++;
```

```
} while( $i < 6 );
```

```
/**
```

- 0

- 1

- 2

- 3

- 4

- 5

*/

Exemple 2

```
$list = [ 'C', 'o', 'd', 'e', 'u', 'r' ];
```

```
$i = 0;
```

```
do {
```

```
    echo "{$list[$i]}.";
```

```
    $i++;
```

```
} while( $i < 6 );
```

```
/** C.o.d.e.u.r. */
```

Boucle for en PHP

On rentre maintenant dans une boucle beaucoup plus intéressante : la boucle for.

La syntaxe est un peu plus compliquée que les précédentes, mais son utilisation est plus simple. La grosse différence est que la déclaration de la variable de comptage ne se fait pas avant la boucle ; l'incrémentation ne se fait pas dans l'action ; les deux se gère dans la déclaration de la boucle.

La syntaxe est la suivante : `for(variation de comptable ; condition ; incrémentation) { action }`

Pour notre premier exemple, la variable de comptage s'appelle \$i et commence à 0. La condition est que \$i doit être inférieur à 6 ; et à chaque passage de la boucle, \$i est incrémenté de 1. Ce qui nous donne : `for($i = 0 ; $i < 6 ; $i++)`.

Exemple 1

```
for( $i = 0 ; $i < 6 ; $i++ ) {
```

```
    echo "- {$i}<br />";
```

```
}
```

```
/**
```

- 0

- 1

- 2

- 3

- 4

- 5

*/

Exemple 2

```
$list = [ 'C', 'o', 'd', 'e', 'u', 'r' ];
```

```
for( $i = 0 ; $i < count($list) ; $i++ ) {  
  
    echo "{$list[$i]}."  
  
}  
  
/** C.o.d.e.u.r. **/
```

Boucle foreach en PHP

Il nous reste le dernier type de boucle : foreach. Elle est assez différente de toutes les autres. « For each » signifiant « Pour chacun », on peut comprendre que cette boucle PHP va boucler sur un élément, en l'occurrence un tableau, pour traiter chacune de ces valeurs.

Il y a deux façons de l'écrire. Dans un premier temps, on va déclarer la boucle avec foreach et des parenthèses. À l'intérieur de celles-ci, on commencera par appeler notre tableau, suivi du mot « as ». Pour la suite, et toujours dans les parenthèses, tout dépend de ce que l'on veut faire. Si on souhaite ne récupérer que les valeurs du tableau, dans ce cas, il suffit de nommer cette valeur avec un nouveau de variable, par exemple \$value. Si par contre on souhaite récupérer l'index, il faudra rajouter avant la variable correspondant à l'index, par exemple, \$key, suivi d'une flèche « => ».

Une boucle foreach peut donc s'écrire foreach(\$list as \$key => \$value) {} ou foreach(\$list as \$value) {}.

Exemple 1

La boucle foreach ne fonctionne qu'avec un tableau ; il n'est donc pas possible de l'utiliser avec le premier exemple.

Exemple 2

```
$list = [ 'C', 'o', 'd', 'e', 'u', 'r' ];  
  
foreach( $list as $key => $value ) {  
  
    echo "{$value}."  
  
}  
  
/** C.o.d.e.u.r. **/
```

Références Bibliographiques

[1] <https://www.codeur.com/tuto/php/boucle-php/>

Last modified: Tuesday, 9 May 2023, 10:06 PM