

# **Les conditions et les boucles**

## **en PHP**

# Les conditions

- Les conditions permettent à PHP de prendre des décisions en fonction de la valeur des variables.
- La forme de condition la plus courante est IF- ELSEIF - ELSE qui signifie « si »... « sinon si »... « sinon ».
- On peut combiner les conditions avec les mots-clés AND ( et ) et OR ( ou )
- Si une condition comporte de nombreux ELSEIF, il peut être plus pratique d'utiliser SWITCH, une autre forme de condition.

# La structure if et else

```
1 <?php
2 $age = 8;
3
4 if ($age <= 12) // SI l'âge est inférieur ou égal à 12
5 {
6     echo "Salut gamin ! Bienvenue sur mon site !<br />";
7     $autorisation_entrer = "Oui";
8 }
9 else // SINON
10 {
11     echo "Ceci est un site pour enfants, vous êtes trop vieux pour pouvoir entrer. Au revoir
12     !<br />";
13     $autorisation_entrer = "Non";
14 }
15 echo "Avez-vous l'autorisation d'entrer ? La réponse est : $autorisation_entrer";
16 ?>
```

Symbole	Signification
==	Est égal à
>	Est supérieur à
<	Est inférieur à
>=	Est supérieur ou égal à
<=	Est inférieur ou égal à
!=	Est différent de

Mot-clé	Signification	Symbole équivalent
AND	Et	&&
OR	Ou	

A  
connaître

# Les boucles

- Les boucles demandent à PHP de répéter des instructions plusieurs fois.
- Boucle while : à utiliser de préférence lorsqu'on ne sait pas par avance combien de fois la boucle doit être répétée
- Boucle for : à utiliser lorsqu'on veut répéter des instructions un nombre précis de fois.

# Boucle while

**while** peut se traduire par  
« tant que ».

Ici, on demande à PHP : **TANT  
QUE** `$nombre_de_lignes` est  
`<=` à 100, exécute ces  
instructions.

```
1 <?php
2 $nombre_de_lignes = 1;
3
4 while ($nombre_de_lignes <= 100)
5 {
6     echo 'Ceci est la ligne n°' . $nombre_de_lignes . '<br />';
7     $nombre_de_lignes++;
8 }
9 ?>
```

**Les instructions qui sont  
répétées en boucle se trouvent  
entre les accolades{et}.**

# Boucle for

**FOR est bien adapté quand on doit compter le nombre de fois que l'on répète les instructions**

**et il permet de ne pas oublier de faire l'incrémentation pour augmenter la valeur de la variable**

```
1 <?php
2 for ($nombre_de_lignes = 1; $nombre_de_lignes <= 100; $nombre_de_lignes++)
3 {
4     echo 'Ceci est la ligne n°' . $nombre_de_lignes . '<br />';
5 }
6 ?>
```

Le premier element sert à l'initialisation. C'est la valeur que l'on donne au départ à la variable (ici, elle vaut 1).

Le second, c'est la condition. Comme pour le while, tant que la condition est remplie, la boucle est réexécutée. Dès que la condition ne l'est plus, on en sort.

Enfin, le troisième c'est l'incrémentation, qui vous permet d'ajouter 1 à la variable à chaque tour de boucle.