

LES BASES DE PHP

3 – Les Tableaux

Table des matières

Les variables de type « array ».....	1
Les tableaux indexés.....	1
Les tableaux associatifs.....	2
Opérateurs dédiés aux tableaux.....	4
Fonctions utiles pour la gestion des tableaux.....	5

Les variables de type « array »

Ces variables stockent des données sous forme de tableau. L'intérêt principal de ce type de variable est de pouvoir stocker plusieurs valeurs (texte ou valeur numérique) dans une seule et même variable. Avec ce type de variable, il devient très simple de trier des valeurs, les organiser, trouver la valeur la plus grande, etc ...

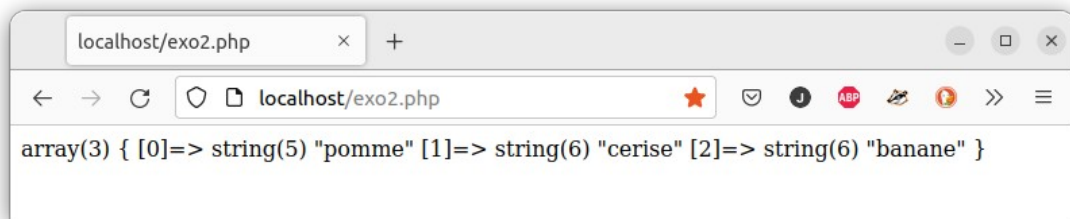
PHP gère 3 types de tableau :

- **les tableaux indexés** (Indexed arrays) : Chaque valeur enregistrée du tableau est associé à un numéro d'index. Le premier élément possède l'index 0, le second l'index 1 , etc .
- **Les tableaux associatif** (Associative arrays) : Les données sont stockés sous la forme de clé=valeur. Pour afficher une valeur, il suffit d'appeler sa clé.
- **Les tableaux multidimensionnels** (Multidimensional arrays) – Ce type de tableau permet de stocker non pas des valeurs, mais des tableaux. C'est donc un tableau de tableaux. Ce point ne sera abordé. Reportez vous à la documentation pour plus de détails.

Les tableaux indexés

Dans l'exemple suivant `$fruits` est un tableau sont stocké les chaînes de caractères « pomme », « cerise » et « banane ». La fonction PHP `var_dump()` renvoie, le nombre d'entrée du tableau entre {}, puis pour chaque élément le n° d'index entre [], le type de donnée, la longueur du champs entre () et sa valeur :

```
<?php
$fruits = array("pomme", "cerise", "banane");
var_dump($fruits);
?>
```



La fonction `var_dump()` est un super outil de débogage. N'hésitez à vous en servir pendant vos phases de test.

Les tableaux associatifs

Les tableaux associatifs utilisent la notion de clé:valeur. Ce mécanisme permet d'identifier une valeur grâce à sa clé. Le couple clé:valeur est donc ce qui constitue une entrée dans un tableau associatif. Le terme associatif illustre le fait qu'une valeur est toujours associée à une clé.

Pour créer un tableau associatif, il existe 2 méthodes. La première :

```
<?php
$age = array("Denis"=>"35", "Pierre"=>"37", "Marie"=>"24");
var_dump($age);
?>
```

La deuxième :

```
<?php
$age['Denis'] = "35";
$age['Pierre'] = "37";
$age['Marie'] = "24";
var_dump($age);
?>
```

Comme expliqué plus haut pour afficher une valeur, il faut mentionner la clé qui lui est associée. Affichons la valeur associée à la clé 'Denis' du tableau associatif `$age`.

```
<?php
// Créons un tableau associatif $age
$age = array("Denis"=>"35", "Pierre"=>"37", "Marie"=>"24");

// affichons la valeur associée de la clé 'Denis' du
// tableau $age.
echo "Denis a " . $age['Denis'] . " ans.";
?>
```

Notez qu'un tableau peut stocker des valeurs de type différents. Cela veut dire que vous pouvez stocker dans un tableau à votre convenance des chaînes de caractères, des entiers, des floats, des booléens, etc ...

```
<?php
$tableau['Denis'] = 24;           // un entier
$tableau['Serge'] = NULL;        // NULL (vide)
$tableau['Pierre'] = "Basket";   // Une chaîne de caractères
$tableau['Marie'] = TRUE;        // Une booléen

var_dump($tableau);
?>
```

En revanche, si une clé est mentionnée 2 fois, la première valeur sera écrasée, comme c'est le cas ici pour la clé « Marie ». PHP considère que si vous faite référence à une clé qu'il connaît déjà et c'est que vous souhaitez modifier cette valeur.

```
<?php
$tableau['Marie'] = TRUE;        // Une booléen
$tableau['Marie'] = 12.241;      // La précédente valeur a été
                                // écrasée

var_dump($tableau);
?>
```

Autre point important, faite attention à vos noms de clé. PHP accepte que vous mentionniez un autre type de donnée que int ou str pour le nom de votre clé. Mais tentera de rendre cohérent ces noms de clé, quitte à tronquer vos valeurs de clé. Prenons un exemple

```
<?php
$tableau['Denis'] = 82,524;

var_dump($tableau);
?>
```

PHP accepte que l'on aie des types de **clé** différent dans tableau associatif. Je peux mentionner une **clé** de type entier (int). Mais je peux aussi lui mettre la valeur 8,15 qui est un float. 8,15 n'étant pas un entier PHP va tronquer le type float en int, c'est à dire 8 ne gardant que la partie entière de ma valeur 8,15. Ce qui revient à modifier ou créer la valeur de l'index n°8 de mon tableau.

```
<?php
$tableau[8] = "test";
$tableau[8.15] = 82.524;

var_dump($tableau);
?>
```

Comme on le voit, ce mécanisme peut provoquer un comportement inattendu s'il n'est pas maîtrisé ou si une faute de frappe c'est glissée dans votre code.

Prenons un autre exemple, un tableau avec un élément possédant une clé 1(int). A cette clé est associé la valeur 12,241 (float).

```
<?php
$tableau[1] = 12.241;
```

```
var_dump($tableau);
?>
```

Ajoutons cette ligne

```
<?php
$tableau[1] = 12.241;
$tableau[true] = 'Salut mon pote !';
var_dump($tableau);
?>
```

A priori, il s'agit d'une entrée supplémentaire à mon tableau. Pourtant quand nous exécutons ce code, une seule entrée est affichée. Que c'est-il passé cette fois ?

Nous avons mentionné comme clé la valeur `TRUE`, soit vrai. PHP n'acceptant que des clé de type entier (INT) ou des chaîne de caractères (STR), l'interpréteur a converti ma clé de type `BOOL` en `TYPE` entier (INT). `TRUE` prendra la valeur de 1 (`FALSE` aurait été converti en 0). Sauf que ici nous avons déjà une clé qui possède la clé 1. Dans ce cas, le comportement normal de PHP sera d'écraser valeur 12,241 (float) par 'Salut mon pote !' ; ce qui n'était pas du tout notre souhait au départ. C'est toujours comme ça quand une connaissance qu'on a pas invité se pointe à sa soirée. C'est le bazar ... Sinon vous avez notez qu'il y avait eu aussi un changement de type de donnée pour la partie « valeur » ... C'est bien, on progresse .

Opérateurs dédiés aux tableaux

Ces opérateurs permettent de réaliser des opérations sur des tableaux comme unir 2 tableaux ou vérifier si 2 tableaux sont identiques ou pas.

Opérateurs	Nom	Exemple	Résultat
+	Union	<code>\$x + \$y</code>	Union de \$x et \$y
==	Égalité	<code>\$x == \$y</code>	Renvoie <code>TRUE</code> si \$x et \$y possède les mêmes paires clé/valeur
===	Identique	<code>\$x === \$y</code>	Renvoie <code>TRUE</code> si \$x et \$y possède les mêmes paires clé/valeur dans le même ordre et de même type
!=	Inégalité	<code>\$x != \$y</code>	Renvoie <code>TRUE</code> si \$x n'est pas égal à \$y
<>	Inégalité	<code>\$x <> \$y</code>	Renvoie <code>TRUE</code> si \$x n'est pas égal à \$y
!==	Non-identique	<code>\$x !== \$y</code>	Renvoie <code>TRUE</code> si \$x n'est pas identique à \$y

Exemple d'union de 2 tableaux

```
<?php
$tableau1['Denis'] = 24;
$tableau1['Serge'] = 28;
$tableau1['Pierre'] = 22;

$tableau2['pomme'] = 257;
$tableau2['abrico'] = 456;
$tableau2['peche'] = 719;

$tableau3 = $tableau1 + $tableau2 ; // Union de 2 Tableaux
var_dump($tableau3);
?>
```

Exemple de comparaison entre 2 tableaux

```
<?php
$tableau1['Denis'] = 24;
$tableau1['Serge'] = 32;
$tableau1['Pierre'] = 21;
$tableau1['Marie'] = 26;

$tableau2['pomme'] = 37;
$tableau2['abricot'] = 9;
$tableau2['peche'] = 22;
$tableau2['orange'] = 56;

var_dump($tableau1 == $tableau2);
?>
```

Les deux tableaux possèdent des clé/valeur différentes, `var_dump($tableau1 == $tableau2)` renvoie `FALSE`, donc faux.

Fonctions utiles pour la gestion des tableaux

La fonction `count ()` est utilisée pour renvoyer la longueur (le nombre d'éléments) d'un tableau :

```
<?php
$arbres = array("Chêne", "Poirier", "Peuplier");
echo count($arbres); // affiche 3 parce que nous avons 3 éléments
                    // dans notre tableau
?>
```

Dans ce chapitre, nous passerons en revue les fonctions de tri de tableau PHP suivantes :

- `sort ()` pour trier les tableaux par ordre croissant
- `rsort ()` pour trier les tableaux par ordre décroissant
- `asort ()` pour trier les tableaux associatifs par ordre croissant, selon la valeur
- `ksort ()` pour trier les tableaux associatifs par ordre croissant, selon la clé
- `arsort ()` pour trier les tableaux associatifs par ordre décroissant, selon la valeur
- `krsort ()` pour trier les tableaux associatifs par ordre décroissant, selon la clé

L'exemple suivant trie les éléments du tableau `$nombres` dans l'ordre croissant (+ vers le -):

```
<?php
$nombres = array(7, 38, 2, 22, 72);
sort($nombres); // trie les éléments par ordre croissant
?>
```

L'exemple suivant trie les éléments du tableau \$nombre dans l'ordre décroissant (du – vers le +):

```
<?php
$nombre = array(7, 38, 2, 22, 72);
rsort($nombre); // trie les éléments par ordre décroissant
var_dump($nombre);
?>
```

L'exemple suivant trie par ordre croissant les valeurs un tableau associatif:

```
<?php
$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");
asort($age); // trie par ordre croissant les valeurs d'un tableau
              //associatif
var_dump($age);
?>
```

L'exemple suivant trie par ordre croissant les clés un tableau associatif:

```
<?php
$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");
ksort($age); // trie par ordre croissant les clés d'un tableau
              //associatif
var_dump($age);
?>
```

L'exemple suivant trie par ordre décroissant les valeurs un tableau associatif:

```
<?php
$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");
arsort($age); // trie par ordre décroissant les valeurs d'un
              // tableau associatif
var_dump($age);
?>
```

L'exemple suivant trie par ordre décroissant les clés un tableau associatif:

```
<?php
$age = array("Peter"=>"35", "Ben"=>"37", "Joe"=>"43");
krsort($age); // trie par ordre décroissant les clés d'un tableau
              // associatif
var_dump($age);
?>
```

A rendre

Calculez en PHP la formule suivante :

- 1) A l'aide d'un tableau, trié par ordre décroissant la suite de Fibonacci
0 ,1 ,1 ,2 ,3 ,5 ,8 ,13 ,21 ,34 ,55 ,89 ,144 , 233 , 377
- 2) Créer un tableau associant à chaque mois de l'année le nombre de jours du mois. (On supposera que l'année n'est pas bissextile.)
 - Quelles sont alors les clés?
 - Les valeurs?
 - Afficher, sous forme de tableau, le nombre de jours de chaque mois.
- 3) Vous devez trouver le moyen de stocker ces données dans un tableau php. En suite afficher uniquement la valeur de l'email de Denis Roger.

Nom	Prénom	Mail	Téléphone
Denis	Roger	Denis.roger@serveur.io	0102030450
Fred	Pasteur	Fred.pasteur@serveur.io	0102030558
Maris	Stuart	Marie.stuart@serveur.io	0102030608