



Tableaux

Technologies web



Introduction aux tableaux PHP

Tableaux

Les tableaux

- En plus des **structures de données** dites « **simples** » (**variables** et **constantes**), PHP propose des types de **données structurées (tableaux)**
- Notez qu'un **tableau PHP** et un **tableau HTML** sont deux choses complètement différentes et ne doivent pas être confondues
- Un **tableau PHP** est chargé de **stocker et manipuler des données** alors qu'un **tableau HTML** sert uniquement à la **mise en forme** d'informations sur une page web
- La force des **tableaux** réside dans le fait que, contrairement aux **types primitifs**, ils peuvent **stocker une ou plusieurs variables** à la fois (possiblement de types différents)

Les tableaux

- Les **tableaux PHP** sont dits « **dynamiques** » car ils s'agrandissent à mesure que vous y ajoutez de nouveaux éléments - Il est donc inutile de préciser sa dimension ni le type de données qu'il contiendra lors de sa déclaration
- **PHP propose 2 types de tableaux :**
 - Les **tableaux à index numériques**
 - Les **tableaux associatifs**

Déclarer un tableau PHP en quelques mots...

- Un **tableau** se déclare de la même manière qu'une **variable** => avec le symbole `$` et un **nom** respectant les mêmes règles de **déclaration** qu'une **variable** traditionnelle
- Bien entendu, un **tableau** est identifié dans notre code par le nom que nous lui attribuons
- Pour **déclarer** un nouveau **tableau**, PHP propose la **fonction** `array()` prenant en **arguments** facultatifs les valeurs à insérer dans le **tableau** à son **initialisation** séparés par des virgules

Déclarer un tableau PHP en pratique...

```
10  <?php
11      // Déclaration d'un tableau vide
12      $fruits = array();
13
14      // Déclaration d'un tableau à index numériques
15      $legumes = array("carotte", "poivron", "aubergine", "chou");
16
17      // Déclaration d'un tableau associatif
18      $identite = array(
19          "prenom" => "Chris",
20          "nom" => "Chevalier",
21          "age" => 28,
22          "estFormateur" => true
23      );
24  ?>
```

Ajouter une valeur à un tableau PHP

- Voici comment ajouter une valeur **dynamiquement** à la fin des **tableaux** déclarés précédemment :

```
$legumes[] = "salade";  
$identite["taille"] = 173;
```

- Il est également possible d'ajouter une valeur à un **index** précis d'un **tableau à index numériques** :

```
$legumes[12] = "endive";  
$legumes[20] = "piment";
```




Les types de tableaux

Tableaux

Les tableaux indexés numériquement

- Un **tableau à index numériques** représente une **liste** d'éléments repérés par un **index** numérique unique
- Le **premier élément** du tableau est situé à l'indice **0**, le **second** à l'indice **1**, le **troisième** à l'indice **2**, etc.
- Ainsi, pour accéder à un élément précis d'un **tableau à index**, on y fait référence avec cette syntaxe : `$monTableau[0]`, `$monTableau[1]`, `$monTableau[2]`

```
10  <?php
11      // Déclaration d'un tableau à index numériques
12      $legumes = array("carotte", "poivron", "aubergine", "chou");
13
14      // Affiche 'aubergine'
15      echo $legumes[2];
16  ?>
```

Les tableaux associatifs

- Le **tableau associatif** est apparu pour pallier les faiblesses des **tableaux indexés numériquement**
- En effet, il peut arriver de ne pas connaître l'emplacement exact d'un élément souhaité dans un **tableau à index numériques**
- De plus, il est moins intuitif de repérer un élément via un **index** que par une **clé**
- Un **tableau associatif** est composé de paires **clé/valeur** - Chaque **clé** fait référence à une **valeur**
- Ainsi, pour accéder à l'une des valeurs du **tableau**, il suffit d'y faire référence =>
`$monTableau['clé']`

Les tableaux associatifs

- Pour récupérer et afficher les différentes valeurs du **tableau associatif** précédemment déclaré, nous écrirons :

```
10  <?php
11      // Déclaration d'un tableau associatif
12      $identite = array(
13          "prenom" => "Chris",
14          "nom" => "Chevalier",
15          "age" => 28,
16          "estFormateur" => true
17      );
18
19      $identite["taille"] = 173;
20
21      echo "Prénom : {$identite["prenom"]} <br>"; // Affiche 'Prénom : Chris'
22      echo "Nom : {$identite["nom"]} <br>"; // Affiche 'Nom : Chevalier'
23      echo "Âge : {$identite["age"]} ans <br>"; // Affiche 'Âge : 28 ans'
24      echo "Taille : {$identite["taille"]} cm <br>"; // Affiche 'Taille : 173 cm'
25  ?>
```

Les tableaux multidimensionnels (1/2)

- Jusqu'alors, nous avons vu comment créer des **tableaux à une seule dimension** - On les appelle également des « **vecteurs** »
- En **PHP**, il est également possible de créer des **tableaux à plusieurs dimensions** - On les appelle « **tableaux de tableaux** »
- Ainsi, pour déclarer une **matrice (tableau à 2 dimensions)** de **3 lignes** par **3 colonnes** pour représenter un jeu de **morions**, nous nous y prendrons ainsi

```
10  <?php
11      // Déclaration d'une matrice 3 x 3
12      $matrice = array();
13      $matrice[0] = array('X', '0', 'X');
14      $matrice[1] = array('X', 'X', '0');
15      $matrice[2] = array('X', '0', '0');
16  ?>
```

Les tableaux multidimensionnels (2/2)

- Ici, nous associons à chaque **index numérique** (**ligne du tableau**) un nouveau **tableau de 3 cases** représentant les **3 colonnes de la ligne**
- Ainsi, si nous souhaitons accéder à la **case centrale** de notre jeu de morpions, on se rend à la **seconde ligne (index 1)** et à la **seconde colonne (index 1)**

```
echo $matrice[1][1]; // Affiche 'X'
```

- On accède donc à une **valeur précise** d'une **matrice** en suivant la syntaxe
`$maMatrice[numeroDeLigne][numeroDeColonne]`

Un tableau spécial : la chaîne de caractères

- En **PHP**, créer une **variable** contenant une **chaîne de caractères** revient à déclarer un **tableau** contenant **autant de cases que de caractères**
- De cette manière, il est possible d'accéder simplement à une **lettre précise** d'une **chaîne de caractères**

```
10  <?php
11      // Déclaration d'un chaîne de caractères
12      $chaine = "Hello World!";
13
14      echo $chaine[6]; // Affiche 'W'
15  ?>
```




Le parcours de tableaux

Tableaux

Parcourir un tableau (1/4)

- En programmation, le **parcours de tableau** se fait par l'utilisation de **boucles**
- Depuis sa **version 4**, **PHP** dispose justement de sa propre **fonction** pour parcourir le contenu d'un **tableau**
- En effet, la **fonction** `foreach()` est une **boucle** particulière avançant le **pointeur** du tableau à chaque **itération** et présente 2 possibilités de syntaxe

Parcourir un tableau (2/4)

```
11 // Affiche chacun des légumes du tableau '$legumes'
12 foreach ($legumes as $legume) {
13     echo "{$legume}<br>"; // Affiche 'carotte', puis, 'poivron', etc.
14 }
15
16 // Affiche chacun des couples clé/valeur du tableau '$identite'
17 foreach ($identite as $cle => $valeur) {
18     echo "{$cle} : {$valeur}<br>"; // Affiche 'prenom : Chris', puis 'nom : Chevalier', etc.
19 }
```

Parcourir un tableau (3/4)

- Cette **fonction** prend en **argument** le **nom du tableau** à parcourir ainsi que les **données** qu'il faut récupérer (les **valeurs** uniquement ou les **clés** et les **valeurs**)
- Avec la première syntaxe, la **variable** `$legume` prend la **valeur de l'élément courant** parcouru (**carotte**, puis **poivron**, etc.)
- Dans la seconde syntaxe, la **variable** `$cle` prend la valeur de la **clé courante** parcourue (**prenom**, puis **nom**, etc.) tandis que la **variable** `$valeur` prend la **valeur associée à cette clé** (**Chris**, puis **Chevalier**, etc.)

Parcourir un tableau (4/4)

- Notez qu'il est également possible de parcourir un **tableau PHP** avec un autre type de **boucle** - Prenons l'exemple d'une boucle `for()`

```
11 // Stocke la taille du tableau '$legumes' dans la variables '$nombreLegumes'
12 $nombreLegumes = sizeof($legumes);
13
14 // Parcours du tableau '$legumes'
15 for ($i = 0; $i < $nombreLegumes; $i++) {
16     echo "{$legumes[$i]} <br>"; // Affiche 'carotte', puis 'poivron', etc.
17 }
```



Compléments

Tableaux

Afficher le contenu d'un tableau PHP

- Pour aider les développeurs dans le **débogage** de leurs programmes **PHP**, la fonction `print_r()` permet d'**afficher le contenu d'un tableau**

```
20      echo "<pre>";
21      print_r($identite);
22      echo "</pre>";
```

- Ici, les balises HTML `<pre></pre>` ne sont pas obligatoires mais permettent de conserver l'**indentation**

```
Array
(
    [prenom] => Chris
    [nom] => Chevalier
    [age] => 28
    [estFormateur] => 1
    [taille] => 173
)
```

Les opérations sur les tableaux PHP

- PHP propose une série de **fonctions natives** permettant de manipuler les **tableaux** :
 - `count()` et `sizeof()` retournent la **taille** du tableau passé en **paramètre**
 - `sort()` trie les éléments du tableau passé en argument **du plus petit au plus grand**
 - `rsort()` trie les éléments du tableau passé en argument **du plus grand au plus petit**
 - `in_array()` permet de **vérifier l'existence d'une valeur** dans un tableau
 - `array_rand()` extrait une ou plusieurs valeurs du tableau sélectionnées au hasard
- Toutes les **fonctions** relatives aux **tableaux** sont accessibles à [cette adresse](#)



TP1 - Les tableaux PHP

Tableaux

1 - Un tableau de fruits

- Dans la section `<?php ?>` d'un fichier `tp1.html`
- Créez une variable `$fruits` qui contiendra un **tableau PHP** avec les valeurs `"mango"`, `"lemon"` et `"blueberry"`
- Affichez `$fruits` dans le **navigateur**
- Bonus : Essayez la **fonction PHP** `var_dump($fruits)`
- Que constatez-vous ? Écrivez votre réponse en commentaire de votre **code PHP**

2 - Accès aux éléments d'un tableau

- Créez une **variable** `$ingredients` contenant un **tableau** avec les valeurs `"eggs"`, `"milk"` et `"butter"`
- Faites apparaître `"milk"` dans le **navigateur**
- Faites apparaître l'**indice** de `"butter"` dans le **navigateur**

3 - Ajouter et supprimer

- Créez une **variable** `$objects` contenant un **tableau** avec les valeurs `"pen"`, `"book"` et `"lamp"`
- Ajoutez `"chair"` au début du **tableau** puis affichez le résultat
- Supprimez `"lamp"` puis affichez le résultat
- Ajoutez `"laptop"` à la fin du **tableau** puis affichez le résultat
- Supprimez `"chair"` puis affichez le résultat

4 - Ordre

- Créez une **variable** `$numbers` qui contiendra un **tableau** avec les valeurs `4`, `10`, `8`, `12` et `6`
- **Inversez** l'**ordre** des éléments de `$numbers` puis affichez le résultat
- Rangez les éléments de `$numbers` dans l'**ordre croissant** puis affichez le résultat



TP - Mise en pratique

Tableaux

TP - Mise en pratique (1/3)

- Dans un fichier `languages.php`, déclarez une **variable** `$languages` chargée de stocker un **tableau à index numériques** contenant plusieurs langages informatiques (HTML, CSS, PHP, JavaScript, etc.) (Cf. [diapo](#))
- Parcourez ce tableau pour afficher un message du type « `Voici vos <nombre de langages> langages informatiques préférés : <langage-1> <langage-2> ...` » (Cf. [diapo](#)) - Pour afficher le nombre de langages stockés dans le **tableau PHP**, utilisez une **fonction PHP** (Cf. [diapo](#))
- Affichez un nouveau message du type « `Parmi ces langages, le <langage> est votre favoris !` » - Il vous faudra récupérer la valeur stockée à un index précis de votre tableau `$languages` (Cf. [diapo](#))

TP - Mise en pratique (2/3)

- Déclarez une nouvelle **variable** `$frameworks` chargée de stocker un **tableau associatif** contenant des langages informatiques en **clé** et des frameworks associés en **valeur** (`HTML => WordPress, CSS => Bootstrap, PHP => Symfony, JavaScript => React`, etc.) (Cf. [diapo](#))
- Parcourez ce tableau pour afficher un message du type « `Le framework <framework-1> est écrit en <langage-1>.` » pour chaque élément parcouru (Cf. [diapo](#))
- Vous afficherez le nom des frameworks **en gras** (``)

TP - Mise en pratique (3/3)

- Voici une proposition d'affichage final attendu pour valider le TP :

Voici vos 3 langages informatiques préférés : HTML CSS PHP
Parmi ces langages, le CSS est votre favoris !
Le framework **WordPress** est écrit en HTML
Le framework **Bootstrap** est écrit en CSS
Le framework **Symfony** est écrit en PHP



★ TP Bonus

Tableaux

★ - Bonus I

- Créez un **tableau** avec le prénom de chacun de vos camarades de promo
- Affichez aléatoirement un prénom tiré au sort

Des questions ?