

Les Fonctions PHP pour les Tableaux

=====

Table des matières

1. Fonctions de création et conversion
2. Fonctions d'ajout et suppression
3. Fonctions d'accès et recherche
4. Fonctions de tri
5. Fonctions de traitement
6. Fonctions de fusion et découpage
7. Fonctions utilitaires

1. Fonctions de création et conversion

`array()` : Crée un nouveau tableau

Exemple : `$fruits = array("pomme", "banane", "orange");`

`range()` : Crée un tableau contenant un intervalle

Exemple : `$nombres = range(1, 5); // [1, 2, 3, 4, 5]`

`explode()` : Convertit une chaîne en tableau

Exemple : `$fruits = explode(",", "pomme,banane,orange");`

`implode()` / `join()` : Convertit un tableau en chaîne

Exemple : `$chaîne = implode(" ", ["pomme", "banane"]);`

2. Fonctions d'ajout et suppression

`array_push()` : Ajoute des éléments à la fin

Exemple : `array_push($fruits, "kiwi", "mangue");`

`array_pop()` : Supprime et retourne le dernier élément

Exemple : `$dernier = array_pop($fruits);`

`array_unshift()` : Ajoute des éléments au début

Exemple : `array_unshift($fruits, "fraise");`

`array_shift()` : Supprime et retourne le premier élément

Exemple : `$premier = array_shift($fruits);`

3. Fonctions d'accès et recherche

`array_key_exists()` : Vérifie l'existence d'une clé

Exemple : `array_key_exists("id", $tableau);`

`in_array()` : Vérifie l'existence d'une valeur

Exemple : `in_array("pomme", $fruits);`

`array_search()` : Recherche une valeur et retourne sa clé

Exemple : `$cle = array_search("pomme", $fruits);`

`array_keys()` : Retourne toutes les clés

Exemple : `$cles = array_keys($tableau);`

`array_values()` : Retourne toutes les valeurs

Exemple : `$valeurs = array_values($tableau);`

4. Fonctions de tri

`sort()` : Tri croissant (modifie le tableau)

Exemple : `sort($nombres);`

`rsort()` : Tri décroissant

Exemple : `rsort($nombres);`

`asort()` : Tri associatif par valeurs

Exemple : `asort($ages);`

`ksort()` : Tri associatif par clés

Exemple : `ksort($ages);`

`arsort()` : Tri associatif décroissant par valeurs

Exemple : `arsort($ages);`

`krsort()` : Tri associatif décroissant par clés

Exemple : `krsort($ages);`

5. Fonctions de traitement

`array_map()` : Applique une fonction à chaque élément

Exemple : `array_map(function($n) { return $n * 2; }, $nombres);`

`array_filter()` : Filtre les éléments selon un critère

Exemple : `array_filter($nombres, function($n) { return $n > 0; });`

`array_reduce()` : Réduit le tableau à une seule valeur

Exemple : `array_reduce($nombres, function($acc, $n) { return $acc + $n; });`

6. Fonctions de fusion et découpage

`array_merge()` : Fusionne plusieurs tableaux

Exemple : `array_merge($tab1, $tab2);`

`array_slice()` : Extrait une portion du tableau

Exemple : `array_slice($tableau, 2, 3);`

`array_splice()` : Modifie une portion du tableau

Exemple : `array_splice($tableau, 2, 1, ["nouveau"]);`

7. Fonctions utilitaires

`array_unique()` : Supprime les doublons

Exemple : `array_unique($tableau);`

`array_reverse()` : Inverse l'ordre des éléments

Exemple : `array_reverse($tableau);`

`array_sum()` : Calcule la somme des valeurs

Exemple : `array_sum($nombres);`

`array_count_values()` : Compte les occurrences des valeurs

Exemple : `array_count_values($tableau);`

`array_diff()` : Trouve les différences entre tableaux

Exemple : `array_diff($tab1, $tab2);`

`array_intersect()` : Trouve les intersections entre tableaux

Exemple : `array_intersect($tab1, $tab2);`

Notes importantes :

- La plupart des fonctions de tri modifient directement le tableau d'origine
- Les fonctions `array_map()`, `array_filter()` et `array_reduce()` sont essentielles pour la programmation fonctionnelle
- Certaines fonctions préservent les clés associatives, d'autres non
- La documentation PHP officielle reste la référence pour plus de détails sur chaque fonction