



8) Les fonctions

Maintenant que nous avons vu les boucles, nous allons voir un nouveau chapitre très important dans les langages de programmation, les fonctions.

Les fonctions comme dans tout langage de programmation permettent d'effectuer une petite partie de logique et de vous renvoyer un résultat. Elle permet d'éviter de se répéter et d'effectuer des traitements spécifiques.

En php les fonctions s'écrivent de la manière suivante : elles ont tout d'abord un nom suivi d'une parenthèse où l'on va mettre des paramètres (0 à plusieurs séparés par des virgules), ces paramètres peuvent être des variables ou des valeurs directement, ensuite ces fonctions vont renvoyer un résultat.

Nous avons déjà utilisé plusieurs fonctions php, tel que **sizeof()** pour la taille des tableaux ou encore **rand()** pour avoir un chiffre au hasard compris entre 2 nombres prédéfinis ou encore **readline()** etc. Toutes ces fonctions sont disponibles dans la documentation officielle de PHP sur www.php.net dans référence du langage ensuite fonctions. Elles sont regroupées par catégorie.

Ici prenons l'exemple de la fonction **rand()**, si vous allez dans la documentation on y voit ceci

rand

(PHP 4, PHP 5, PHP 7, PHP 8)

rand — Génère une valeur aléatoire

Description

```
rand(): int
```

```
rand(int $min, int $max): int
```

Appelée sans les options **min** et **max**, **rand()** retourne un nombre pseudoaléatoire entre 0 et **getrandmax()**. Si vous voulez un nombre aléatoire entre 5 et 15 (inclus), par exemple, utilisez `rand (5, 15)`.

On peut voir sa syntaxe :

- Elle a un nom **rand**
- Des paramètres entre parenthèses ou pas (**int \$min, int \$max**) ou ()
- Et elle retourne une valeur de type entier via : **int**

Je répète encore une fois, une fonction a un nom de fonction suivi de parenthèses collées, dans ces parenthèses on y met si demander de 0 à plusieurs arguments séparés par des virgules. **Attention pas d'espace entre le nom et les parenthèses.**

- **nomFonction(variable(s)/valeur(s),...)** ;

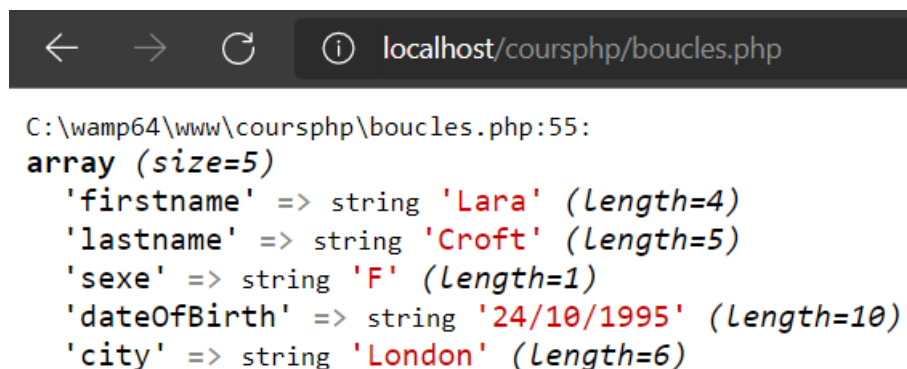
La fonction **var_dump()**

La fonction **var_dump()** est une fonction qui va afficher ce qu'on lui passe en paramètres. Elle sert principalement lors du débogage quand on construit une application et qu'on veut vérifier par moment la valeur d'une variable, un objet, un tableau etc.

Testons-le, créez un fichier **fonctions.php** et introduisez ce code :

```
$tabIndexe = [  
    'firstname' => 'Lara',  
    'lastname' => 'Croft',  
    'sexe' => 'F',  
    'dateOfBirth' => "24/10/1995",  
    'city' => 'London'  
];  
var_dump($tabIndexe);
```

Lancer que ce soit dans le terminal ou dans le navigateur on obtiendra quelque chose de ce style :



The screenshot shows a web browser window with the address bar displaying 'localhost/coursphp/boucles.php'. Below the address bar, the output of the PHP script is shown, indicating the file path 'C:\wamp64\www\coursphp\boucles.php:55:'. The output is an array dump for an associative array with 5 elements. Each element is shown with its key, the value, its type (string), and its length in parentheses. The keys are 'firstname', 'lastname', 'sexe', 'dateOfBirth', and 'city'. The values are 'Lara', 'Croft', 'F', '24/10/1995', and 'London' respectively.

```
C:\wamp64\www\coursphp\boucles.php:55:  
array (size=5)  
    'firstname' => string 'Lara' (length=4)  
    'lastname' => string 'Croft' (length=5)  
    'sexe' => string 'F' (length=1)  
    'dateOfBirth' => string '24/10/1995' (length=10)  
    'city' => string 'London' (length=6)
```

Ça nous détail ce qu'il y a donc dans notre tableau associatif \$tabIndexe.

Il nous affiche à chaque fois le type ainsi que la taille de chacune des données.

Il n'est pas utilisé pour l'affichage dans une application, il sert vraiment à voir si une variable a bien reçu ce qu'elle devait recevoir, on l'utilisera dans les projets qu'on fera plus tard.

(Exos) Vous devez aller sur internet pour trouver les fonctions nécessaires

- 1) Reprenez le tableau de note qu'on a déjà utilisé à plusieurs reprises, et trouvez-moi une fonction qui va recevoir en paramètre ce tableau d'entier, et qui va me ressortir le maximum. Affichez ensuite :

```
← → ↻ ⓘ localhost/coursphp/fonctions.php
```

Votre tableau est : [18 13 5 10 9]

Voici la valeur max : 18

- 2) Je veux une fonction qui vérifie si un mot entré au clavier est un palindrome, c'est-à-dire qu'on sait le lire de droite à gauche ou de gauche à droite. Par exemple « ere, rever, ressasser, kayak, elle, gag etc. ». Essayer de trouver une fonction en php qui permet de pouvoir faire cela. (Astuce comparaison du mot à l'endroit avec le mot inverser). Je veux qu'il affiche 2 messages différents en fonction de si il est palindrome ou pas.

```
PS C:\wamp64\www\coursphp> php fonctions.php
Veuillez entrez un mot : stats
Votre mot est un palindrome et donne stats a l'envers
```

- 3) Reprenez l'exercice 2, si vous l'avez fait quand vous essayez de mettre un mot qui est un palindrome avec une majuscule, vous remarquez qu'il ne fonctionne pas (Case Sensitive donc pour lui Kayak n'est pas la même chose que kayak). Essayer cette fois-ci de trouver une fonction qui permet de remédier à ce problème de majuscule (Astuce on l'a vu en MySQL).
- 4) Reprenez encore une fois le tableau de notes, avant on faisait la moyenne en reprenant dans une boucle élément par élément pour faire la somme, ensuite à la sortie de la boucle on divisait par la taille du tableau. Je vous demande de faire autrement, il y a une fonction qui permet d'avoir la somme des éléments d'un tableau ensuite vous connaissez la fonction qui permet d'avoir la taille d'un tableau. Donnez-moi la moyenne du tableau de notes.
- 5) Je veux que vous trouvez une fonction qui permet d'inverser un tableau. Donc si par exemple j'ai comme tableau le tableau de notes :

```
[18,13,5,10,9]
```

Je veux comme résultat :

```
← → ↻ ⓘ localhost/coursphp/fonctions.php
```

Voici votre tableau [18 13 5 10 9]

Voici votre tableau inversé [9 10 5 13 18]

- 6) Reprenez le même tableau de note et je veux que vous l'affichiez trié. Trouvez la bonne fonction. Trié de manière croissante.
- 7) Cette fois-ci je le veux de manière trié décroissante.