| **Fonction PHP** | **Rôle** |
| --- | --- |
| **date**(string $format, [int $timestamp]) | Renvoie une string de la date $timestamp (par défaut la date actuelle) reformatée selon les règles dans $format |
| **getdate**([int $timestamp]) | Renvoie un tableau associatif (clés => valeurs) contenant les informations de $timestamp (par défaut la date actuelle) |
| **time**(void) | Renvoie le timestamp actuel |
| **trim**($string [, $character\_mask]) | supprime de $string les caractères de la chaine $character\_mask, par défaut les espaces |
| **ltrim**($string [, $character\_mask]) | même comportement que trim() appliqué uniquement au début de la $string (left trim) |
| **rtrim**($string [, $character\_mask]) | même comportement que trim() appliqué uniquement à la fin de la $string (right trim) |
| **strlen**($string) | renvoie la longueur de la $string |
| **strtoupper**($string) | renvoie la conversion en majuscules de la $string |
| **ucfirst**($string) | met en majuscule le premier caractère de la $string et renvoie la chaîne créée |
| **substr**($string, int $start [, int $length]) | renvoie le segment de $string à partir $start de longueur $length (par défaut jusqu'à la fin de $string). Si $start est négatif, le segment commencera à compter à partir de la fin |
| **strpos**($string, $toFind [, $offset]) | Renvoie la position de l'élément $toFind dans $string. $offset permet de définir à partir d'où $string sera parcouru |
| **explode**($separateur, $string [, $limit]) | revoie le tableau des sous-chaînes de $string, séparée par $separateur. Si $limit est défini, renvoie $limit éléments, le dernier étant le reste de la chaîne |
| **hmlspecialchars**($string) | convertit les caractères spéciaux dde $string en entités HTML |
| **urlencode**($string) | renvoie une version encodée de $string insérable dans une URL (plus d'espace, etc) |
| **isset**($var[,$var2 ...]) | renvoie TRUE si toutes les variables passées en paramètres sont définies et différentes de NULL |
| **empty**($var) | renvoie FALSE si $var est définie et non vide |
| **$\_GET** | Tableau associatif (clé => valeur) passé par l'URL (méthode GET). exemple: |
| echo htmlspecialchars($\_GET["name"]); | renvoie Geraud si l'URL était http://example.com/?name=Geraud |
| **$\_POST** | Tableau associatif passé au script par la méthode POST (voir **$\_GET**) |
| **$\_REQUEST** | Un tableau associatif qui contient par défaut le contenu des variables $\_GET, $\_POST et $\_COOKIE. |
| **$\_SERVER** | Tableau associatif des informations du serveur **(à préciser)** |
| **$\_SESSION** | Tableau associatif des valeurs stockées dans la session. La session s'ouvre dès l'entrée sur le site du visiteur, et se ferme dès sa sortie, au bout d'un temps d'inactivité (timeout) ou la déconnexion via session\_destroy(). $\_SESSION nécessite l'appel de la fonction **start\_session**() **AVANT TOUT CODE HTML ET SUR CHAQUE PAGE L'UTILISANT** pour pouvoir être utilisé |
| **$\_COOKIE** | Un tableau associatif de variables, passé au script courant, via des cookies HTTP. |
| **include**($string) | Inclut au script actuel le fichier spécifié par le chemin $string. (exemple un autre script php) |
| **require**($string) | Comportement analogue à include() mais renvoie une erreur fatale (et donc arrête le script) si l'inclusion renvoie une erreur (include n'émet qu'une alerte **E\_WARNING** le cas échéant |
| **include\_once**($string) | idem include(), mais si le fichier a déjà été inclus, ne le réinclut pas et renvoie TRUE |
| **require\_once**($string) | idem include\_once(), pour require() |
| **key\_exists**($key, $array) | alias de **array\_key\_exists**(), vérifie si la clé $key existe dnas $array. renvoie le booléen TRUE si vrai, FALSE sinon |
| **in\_array**($toFind, $array) | renvoie TRUE si $toFind est dans $array, FALSE sinon |
| **array\_search**($toFind, $array) | renvoie la clé associé à $toFind s'il se trouve dans $array, FALSE sinon |
| **array\_keys**($array [,$search\_value]) | renvoie un tableau des clés de $array. Si $search\_value est spécifié, ne renvoie que les clés contenant la valeur de $search\_value |
| **array\_values**($array) | renvoie toutes les valeurs d'un tableau associatif indexé de façon numérique |
| **is\_array**($var) | renvoie TRUE si $var est un tableau, FALSE sinon |
| **sort**($array) | trie le tableau $array dans l'ordre croissant **(à préciser)** |
| **ksort**($array) | trie le tableau $array suivant ses clés |
| **asort**($array) | trie le tableau $array en conservant l'association clé => valeur |
| **each**($array) | renvoie chaque paire clé/valeur d'un tableau |
| **print\_r**($var) | Affiche des informations lisibles pour une variable |