**Các function xử lý array / string**

**ARRAY**

|  |  |  |
| --- | --- | --- |
| STT | Tên hàm | Công dụng |
| 1 | is\_array($array) | Hàm có tác dụng kiểm tra xem một biến có phải mảng hay không. Trả về true nếu là mảng và ngược lại sẽ trả về false. |
| 2 | **in\_array($value,$array)** | Hàm có tác dụng kiểm tra xem mảng $array có tồn tại giá trị $value hay không. Nếu có sẽ trả về true và ngược lại trả về false. |
| 3 | **array\_values($array)** . | Chuyển mảng **$array**sang dạng mảng chỉ mục |
| 4 | **array\_count\_values($array)** | Đếm số lần xuất hiện của các phần tử giống nhau trong mảng **$array** và trả về một mảng kết quả với các value là các giá trị đếm đó. |
| 5 | **array\_keys($array)** | Hàm này có tác dụng trả về một dạng mảng chỉ mục mà trong mảng đó sẽ có phần tử là key của mảng ban đầu. |
| 6 | **array\_change\_key\_case($array, $case)** | Tác dụng của hàm này là chuyển tất cả các **chỉ mục key** trong mảng **$array** sang chữ hoa nếu **$case** = 1 và sang chữ thường nếu **$case** = 0.  Ta có thể dùng hằng số **CASE\_UPPER** thay cho số 1 và **CASE\_LOWER** thay cho số 0. |
| 7 | **array\_key\_exists($key,$array)** | Kiểm tra xem khoá **$key** có **tồn tại** trong mảng**$array** hay không. Nếu có sẽ trả về true và trả về **false** nếu không có. |
| 8 | **array\_pop($array)** | Tác dụng của hàm này sẽ cắt phần tử cuối cùng ra khỏi mảng.  Ta có thể xuất ra phần tử đã bị cắt từ hàm**array\_pop($array)** bằng cách gán giá trị của nó vào một biến và xuất biến đó ra. Như sau: |
| 9 | **array\_push($array,$value1,$value2…)** | Hàm này có tác dụng thêm một hoặc nhiều phần tử vào **cuối mảng** với các các giá trị **$value1, $value2** được truyền vào.  Nếu ta **var\_dump** hàm **array\_push** này thì kết quả màn hình sẽ trả về tổng cộng số lượng phần tử có trong mảng sau khi thêm. |
| 10 | **array\_shift($array)** | Trái ngược với hàm **array\_pop**, hàm **array\_shift** sẽ cắt phần tử đầu tiên trong mảng **$array** ra khỏi mảng. Ta có thể xuất giá trị đã bị cắt khỏi mảng đó bằng cách gán giá trị của **array\_shift($array)** vào 1 biến và xuất nó ra. |
| 11 | **array\_unshift($array,$value1,$value2…)** | Trái ngược với array\_push, hàm array\_unshift sẽ thêm một hoặc nhiều phần tử vào **đầu mảng** với các các giá trị **$value1, $value2** được truyền vào.  Nếu ta **var\_dump** hàm **array\_push** này thì kết quả màn hình sẽ trả về tổng cộng số lượng phần tử có trong mảng sau khi thêm. |
| 12 | **array\_combine($array\_keys, $array\_values)** | Tác dụng của hàm là trộn 2 mảng **$array\_keys** và **$array\_values** thành một mảng kết hợp. Trong đó, **$array\_keys** là danh sách **keys**, **$array\_value** là danh sách value tương ứng với key. Với điều kiện là 2 mảng này phải bằng nhau. |
| 13 | **array\_merge($array,$array…)** | Cao cấp hơn hàm **array\_combine** (chỉ có tác dụng gộp 2 mảng thành 1), hàm **array\_merge** này có tác dụng gộp 2 hoặc nhiều mảng hơn nữa thành 1 mảng duy nhất |
| 14 | **array\_rand($array, $number)** : | Hàm có tác dụng lấy ra key ngẫu nhiên trong mảng với**$number** là số lượng muốn lấy. Mình sẽ lấy ví dụ tiếp tục từ phần **array\_merge** trên |
| 15 | **array\_unique($array)** | Hàm giúp loại bỏ giá trị trùng nếu có trong mảng **$array.** |
| 16 | **array\_flip($array)** | Hàm này có tác dụng chuyển đổi key của mảng thành value và ngược lại. |
| 17 | **array\_reverse($array)** | Hàm có tác dụng đảo ngược lại vị trí sắp xếp của tất cả phần tử có trong mảng. |
| 18 | **array\_search($keyword, $array)** | Hàm có tác dụng tìm kiếm giá trị của mảng và trả về key của phần tử đó nếu có. Trong đó **$keyword** là giá trị truyền vào, **$array** là mảng cần tìm. |
| 19 | **array\_slice($array,$begin,$number)** | Hàm có tác dụng lấy ra số lượng các phần tử được truyền vào thông qua**$number** và bắt đầu lấy ra từ **$begin** trong mảng. |

**String**

|  |  |  |
| --- | --- | --- |
| STT | Tên hàm | Công dụng |
| 1 | **strlen($string)** . | Hàm này có tác dụng đếm xem chuỗi **$string** có bao nhiêu ký tự và xuất ra màn hình độ dài của chuỗi |
| 2 | **strpos($string, $keyword)** | Hàm này có tác dụng kiểm tra xem đoạn ký tự **$keyword** có tồn tại trong chuỗi **$string** hay không và xuất ra màn hình vị trí bắt đầu tồn tại của **$keyword** đó trong chuỗi. |
| 3 | **explode($separator, $string, $limit)** | Hàm này sẽ chuyển một chuỗi **$string** thành một mảng gồm các phần tử là các ký tự trong mảng. Điều kiện tách mảng sẽ là chuỗi ký tự**$separator**, và giới hạn phần tử sẽ là **$limit**. |
| 4 | **implode($separator, $array)** | Hàm này có tác dụng nối tất cả các phần tử của mảng **$array** thành chuỗi với mỗi phần tử cách nhau bằng chuỗi **$separator.** |
| 5 | **str\_word\_count($string)** | Hàm này sẽ đếm tổng số lượng trả về **số từ** trong chuỗi **$string.** |
| 6 | **substr($string,$start,$length)** | Hàm này có tác dụng cắt ra một chuỗi con nằm trong chuỗi **$string** bắt đầu từ ký tự có vị trí thứ **$start** và chiều dài chuỗi con là **$length** Nếu **$start** là **số dương** thì vị trí bắt đầu sẽ tính từ trái sang phải trong chuỗi. Ngược lại, nếu **$start** là **số âm** thì vị trí bắt đầu sẽ được tính từ phải sang trái của chuỗi. |
| 7 | **strstr($string, $start\_string)** . | Tác dụng của hàm này sẽ tách một chuỗi con từ chuỗi**$string** được tính bắt đầu từ $**start\_string** cho đến hết chuỗi cha |
| 8 | **strtolower($string)** | Hàm có tác dụng chuyển chuỗi **$string** sang dạng chữ thường. |
| 9 | **strtoupper($string)** | Hàm có tác dụng chuyển đổi chuỗi **$string** sang dạng in hoa |
| 10 | **strip\_tags($string, $allow)** | Hàm này sẽ loại bỏ toàn bộ các thẻ **html** có trong chuỗi **$string**, trừ các thẻ được cho phép **$allow**. |

**Các quy tắc thay thế:**

* **\n** được thay thế bằng ký tự newline (dòng mới)
* **\r** được thay thế bởi ký tự carriage-return, được hiểu là đưa con trỏ về đầu dòng nhưng không xuống dòng.
* **\t** được thay thế bởi ký tự tab
* **$** được thay thế bằng một dấu $
* **\”** được thay thế bằng một dấu nháy kép “
* **\\** được thay thế bằng một dấu nháy đơn \