

---

# PHP Array 简介

数组函数允许您访问和操作数组。支持单维和多维数组。数组函数属于 PHP 核心部分。无需安装即可使用这些函数。

## PHP 5 Array 函数

函数	描述
<code>array()</code>	创建数组。
<code>array_change_key_case()</code>	把数组中所有键更改为小写或大写。
<code>array_chunk()</code>	把一个数组分割为新的数组块。
<code>array_column()</code>	返回输入数组中某个单一列的值。
<code>array_combine()</code>	通过合并两个数组来创建一个新数组。
<code>array_count_values()</code>	用于统计数组中所有值出现的次数。
<code>array_diff()</code>	比较数组，返回差集（只比较键值）。
<code>array_diff_assoc()</code>	比较数组，返回差集（比较键名和键值）。
<code>array_diff_key()</code>	比较数组，返回差集（只比较键名）。
<code>array_diff_uassoc()</code>	比较数组，返回差集（比较键名和键值，使用用户自定义的键名比较函数）。
<code>array_diff_ukey()</code>	比较数组，返回差集（只比较键名，使用用户自定义的键名比较函数）。
<code>array_fill()</code>	用给定的键值填充数组。
<code>array_fill_keys()</code>	用指定键名的给定键值填充数组。
<code>array_filter()</code>	用回调函数过滤数组中的元素。

array_flip()	交换数组中的键和值。
array_intersect()	比较数组，返回交集（只比较键值）。
array_intersect_assoc()	比较数组，返回交集（比较键名和键值）。
array_intersect_key()	比较数组，返回交集（只比较键名）。
array_intersect_uassoc() )	比较数组，返回交集（比较键名和键值，使用用户自定义的键名比较函数）。
array_intersect_ukey()	比较数组，返回交集（只比较键名，使用用户自定义的键名比较函数）。
array_key_exists()	检查指定的键名是否存在于数组中。
array_keys()	返回数组中所有的键名。
array_map()	把数组中的每个值发送到用户自定义函数，返回新的值。
array_merge()	把一个或多个数组合并为一个数组。
array_merge_recursive()	递归地合并一个或多个数组。
array_multisort()	对多个数组或多维数组进行排序。
array_pad()	用值将数组填补到指定长度。
array_pop()	删除数组的最后一个元素（出栈）。
array_product()	计算数组中所有值的乘积。
array_push()	将一个或多个元素插入数组的末尾（入栈）。
array_rand()	返回数组中一个或多个随机的键。
array_reduce()	通过使用用户自定义函数，以字符串返回数组。
array_replace()	使用后面数组的值替换第一个数组的值。
array_replace_recursive	递归地使用后面数组的值替换第一个数组的值。

()	
array_reverse()	以相反的顺序返回数组。
array_search()	搜索数组中给定的值并返回键名。
array_shift()	删除数组中首个元素，并返回被删除元素的值。
array_slice()	返回数组中被选定的部分。
array_splice()	删除并替换数组中指定的元素。
array_sum()	返回数组中值的和。
array_udiff()	比较数组，返回差集（只比较值，使用一个用户自定义的键名比较函数）。
array_udiff_assoc()	比较数组，返回差集（比较键和值，使用内建函数比较键名，使用用户自定义函数比较键值）。
array_udiff_uassoc()	比较数组，返回差集（比较键和值，使用两个用户自定义的键名比较函数）。
array_uintersect()	比较数组，返回交集（只比较值，使用一个用户自定义的键名比较函数）。
array_uintersect_assoc()	比较数组，返回交集（比较键和值，使用内建函数比较键名，使用用户自定义函数比较键值）。
array_uintersect_uassoc()	比较数组，返回交集（比较键和值，使用两个用户自定义的键名比较函数）。
array_unique()	删除数组中的重复值。
array_unshift()	在数组开头插入一个或多个元素。
array_values()	返回数组中所有的值。
array_walk()	对数组中的每个成员应用用户函数。

---

<code>array_walk_recursive()</code>	对数组中的每个成员递归地应用用户函数。
<code>arsort()</code>	对关联数组按照键值进行降序排序。
<code>asort()</code>	对关联数组按照键值进行升序排序。
<code>compact()</code>	创建包含变量名和它们的值的数组。
<code>count()</code>	返回数组中元素的数目。
<code>current()</code>	返回数组中的当前元素。
<code>each()</code>	返回数组中当前的键 / 值对。
<code>end()</code>	将数组的内部指针指向最后一个元素。
<code>extract()</code>	从数组中将变量导入到当前的符号表。
<code>in_array()</code>	检查数组中是否存在指定的值。
<code>key()</code>	从关联数组中取得键名。
<code>krsort()</code>	对数组按照键名逆向排序。
<code>ksort()</code>	对数组按照键名排序。
<code>list()</code>	把数组中的值赋给一些变量。
<code>natcasesort()</code>	用“自然排序”算法对数组进行不区分大小写字母的排序。
<code>natsort()</code>	用“自然排序”算法对数组排序。
<code>next()</code>	将数组中的内部指针向前移动一位。
<code>pos()</code>	<code>current()</code> 的别名。
<code>prev()</code>	将数组的内部指针倒回一位。
<code>range()</code>	创建包含指定范围单元的数组。
<code>reset()</code>	将数组的内部指针指向第一个元素。

---

<code>rsort()</code>	对数组逆向排序。
<code>shuffle()</code>	将数组打乱。
<code>sizeof()</code>	<code>count()</code> 的别名。
<code>sort()</code>	对数组排序。
<code>uasort()</code>	使用用户自定义的比较函数对数组中的键值进行排序。
<code>uksort()</code>	使用用户自定义的比较函数对数组中的键名进行排序。
<code>usort()</code>	使用用户自定义的比较函数对数组进行排序。