

PHP 实际上并不区分数值数组和键值对数组；当一个数组值未指定键时，默认取此前出现最大整值+1为键；每个数组单元（键值对）后要加一个逗号，最后一组后也可以加以备添加新单元方便。

默认添加的 key 为整数，可以指定的 key 有字符串、整数以及一些会被强制转换为整数的。如 $8.7 \rightarrow 8$ ， $"1" \rightarrow 1$ ， $true \rightarrow 1$ ， $false \rightarrow 0$ ， $null \rightarrow ""$ ；相同

key 有多个值时前者会被后者覆盖；

如: `$array = array (`

`1 => 'a',`

`1.2 => "b",`

`'1' => "c",`

`true => "d",`

`false => "e",`

`);`

`var_dump($array);`

$\Rightarrow \text{array}(1) \{ [1] \Rightarrow \text{string}(1) "d", [0] \Rightarrow \text{string}(1) "c" \}$

键名为字符串时，要加上单引号；对于

不加引号的, 会先查寻是否有同名常量,
有就以常量值为键名, 无则转为字符串;
对双引号则同所有双引号字符串一样,
会对存在的变量解析取值为键名.

e.g. `$a = 'first';`
`define('second', 'third')`

```
$array = array(  
    'first' => 'a',  
    'second' => 'b',  
    second => 'c',  
);
```

var_dump (\$array) ;

→ array (3) { ["first"] ⇒ 'a' , ["second"] ⇒ 'b' ,

["third"] ⇒ 'c' }

↗ 解析
array [\$a] = array [" \$a "] = array [first]

array [second] = array [third] ≠ array [second]

↓
常量值 'third'

\$array [] [] ... 访问多维数组.
↓ ↓
1级 2级

通过 unset (\$array) 或 unset (\$array [])

删除数组或单元. 三不

```

foreach( $array as (key $key => ) $value ) {
    echo $key, '<br/>';
    echo $value;
}

```

来遍历数组，不用区分数组与键值对数组

用 `var_dump($array)` 或 `print_r($array)` 来显示数组内容。

用 `array_values($array)` 对 `unset()` 过的 `$array` 重建索引 (不然 `unset` 的键值虽删除但索引数还存在)

count() 可以数 array 单元数

sort() 可以给数值数组排序