

第5章 数组

学习要点:

- 1.什么是数组
- 2.自定义键数组
- 3.数组里的数组
- 4.数组的排序
- 5.数组的指针操作
- 6.统计数组个数
- 7.将数组转换成标量变量

主讲教师: 李炎恢

合作网站: <http://www.ibeifeng.com>

讲师博客: <http://hi.baidu.com/李炎恢>

一. 什么是数组

传统上把数组(array)定义为一组有某种共同特性的元素, 包括相似性和类型。每个元素由一个特殊的标识符来区分, 称之为键(key); 而每个键对应一个值(value)。

依照上表, 我们创建一个数组。

第一种:

```
$userNames=array('李彦宏','周鸿祎','马云','俞敏洪','李开复','吴祁');
```

这是索引数组初始化:

数字索引的初始值是从0 开始计算的:

`userNames[0]---`userNames[5], 代表这6个人的名字。

第二种:

通过range()函数自动创建一个数组

```
$numbers=range(1,10) $letters=range('a','z')
```

可以加第三个参数`$numbers=range(1,10,2)`, 第三个参数表示步长

访问数组的内容:

要访问一个变量的内容, 可以直接使用其名称。如果该变量是一个数组, 可以使用变量名称和关键字或索引的组合来访问其内容。

`$numbers[0]`、`$numbers[1]`、`$numbers[2]`等。

改变数组的值:

```
$numbers[0]="Lee";
```

李彦宏周鸿祎马云俞敏洪李开复吴祁

使用循环访问数组

由于数组使用有序的数字作为索引, 所以使用一个for 循环就可以很容易地显示数组的内容:

```
for ($i=0;$i<10;$i++) {  
echo $numbers[$i];  
}
```

也可以使用foreach 循环来遍历数组:

```
foreach ($numbers as $value) {  
    echo $value;  
}
```

测试是否为数组变量: is_array();

print_r 函数: 打印关于变量的易于理解的信息

二. 自定义键数组

初始化相关数组

```
$ages=array('吴祁'=>19,'李炎恢'=>27,'胡心鹏'=>23);
```

访问数组元素

```
$ages["吴祁"];
```

```
$ages["李炎恢"];
```

```
$ages["胡心鹏"];
```

追加数组: 首先, 创建只有一个元素的数组, 然后追加两个元素。

```
$ages=array("吴祁"=>19);
```

```
$ages["李炎恢"]=27;
```

```
$ages["胡心鹏"]=23;
```

直接添加数组: 无需创建, 直接添加, 添加第一个元素自动建立数组。

```
$ages["吴祁"]=19;
```

```
$ages["李炎恢"]=27;
```

```
$ages["胡心鹏"]=23;
```

使用循环语句: 因为相关数组的索引不是数字, 因此无法使用for 循环语句中使用一个简单的计数器对数组进行操作。但是可以使用foreach 循环或list()和each()结构。

```
foreach ($ages as $key=>$value) {  
    echo $key."=>".$value."<br />";  
}
```

使用each()结构打印\$ages 数组的内容: each()函数返回数组的当前元素, 并将下一个元素作为当前元素。

```
while (!!$element=each($ages)) {  
    echo $element["key"];  
    echo "=>";  
    echo $element["value"];  
    echo "<br />";  
}
```

使用list()函数, 可以用来将一个数组分解为一系列的值。可以按照如下方式将函数each()返回的两个值分开:

```
list($name,$age)=each($ages);
```

当使用each()函数时, 数组将记录当前元素。如果希望在相同的脚本中两次使用该数组, 就必须使用函数reset()将当前元素重新设置到数组开始处。

```
reset($prices);
```

确定唯一的数组元素: array_unique();它会删除掉里面相同值的元素。

置换数组键和值: array_flip();它会对调数组中的key 和value;

三. 数组里的数组

数组不一定就是一个关键字和值的简单列表----数组中的每个位置用来保存另一个数

组。使用这种方法，可以创建一个二维数组。可以把二维数组当成一个矩阵，或者是一个具有宽度和高度或者行和列的网格。

二维表

```
$products=array(
array("苹果",6,28.8),
array("猪肉",2,32.1),
array("饼干",3,45.3)
);
```

显示这个二维数组：

```
echo "|".$products[0][0].".$products[0][1].".$products[0][2]."<br />";
echo "|".$products[1][0].".$products[1][1].".$products[1][2]."<br />";
echo "|".$products[2][0].".$products[2][1].".$products[2][2]."<br />";
```

此外，还可以使用双重for 循环来实现同样的效果：

```
for ($row=0;$row<3;$row++) {
for ($column=0;$column<3;$column++) {
```

产品名数量价格

苹果6 28.8

猪肉2 32.1

饼干3 45.3

```
echo "|".$products[$row][$column];
}
echo "<br />";
}
```

使用列明的二维数组：

```
$products=array(
array("产品名"=>"苹果","数量"=>6,"价格"=>28.8),
array("产品名"=>"猪肉","数量"=>2,"价格"=>32.1),
array("产品名"=>"饼干","数量"=>3,"价格"=>45.3)
);
```

显示这个二维数组：

第一种方式：

```
for ($row=0;$row<3;$row++) {
echo "|".$products[$row]["产品名"].".$products[$row]["数量"].".$products[$row]["价格"]."<br />";
}
```

第二种方式：

```
for ($row=0;$row<3;$row++) {
while (!!list($key, $value)=each($products[$row])) {
echo "|".$value;
}
echo "<br />";
}
```

四．数组的排序

对保存在数组中的相关数据进行排序是一件非常有意义的事情。使用并且排序一个一维数组是非常简单的。

使用sort()函数将数组按字母升序进行排序。

```
$products=array("orange","banner","apple");
```

```
sort($products);
```

使用sort()函数将数字升序进行排序。

```
$prices=array(100,10,4,23,78);
```

```
sort($prices);
```

sort()函数的第二个参数是可选的。这个可选参数可以传递SORT_REGULAR（默认值）、SORT_NUMERIC 或SORT_STRING。指定排序类型的功能是非常有用的。比如，当要比较可能包含有数字2 和12 的字符串时，从数字角度看，2 要小于12，但是作为字符串，"12" 却要小于"2"。

使用asort()函数和ksort()函数对相关数组排序

创建一个包括3 个产品及价格的数组，然后将它们按价格的升序进行排序：

```
$prices=array("c"=>苹果,"a"=>猪肉,"b"=>饼干);
```

```
asort($prices);
```

按说明排序使用ksort()函数

```
ksort($prices)
```

反向排序：sort()、asort()和ksort()都是正向排序，当然也有相对应的反向排序。

实现反向：rsort()、arsort()和krsort()。

在一些应用程序中，可能希望按另一种方式对数组排序。函数shuffle()将数组个元素进行随机排序。函数array_reverse()给出一个原来数组的反向排序。

```
shuffle($pictures);
```

```
array_reverse($pictures);
```

array_unshift()函数将新元素添加到数组头，array_push()函数将每个新元素添加到数组的末尾。

array_shift()删除数组头第一个元素，与其相反的函数是array_pop(),删除并返回数组末尾的一个元素。

array_rand()返回数组中的一个或多个键。

五. 数组的指针操作

在数组中浏览：each()、current()、reset()、end()、next()、pos()、prev();

调用next()或each()将使指针前移一个元素。调用each(\$array_name)会在指针前移一个位置之前返回当前元素。next()函数则有些不同----调用next(\$array_name)是将指针前移，然后再返回新的当前元素。

要反向遍历一个数组，可以使用end()和prev()函数。prev()函数和next()函数相反。它是将当前指针往回移一个位置然后再返回新的当前元素。

六. 统计数组个数

count()和sizeof()统计数组下标的个数

array_count_values()统计数组内下标值的个数

七. 将数组转换成标量变量：extract()

对于一个非数字索引数组，而该数组又有许多关键字-值对，可以使用函数extract()将它们转换成一系列的标量变量。extract()函数原型如下：

```
extract(array var_array,[int extract_type],[string prefix]);
```

函数extract()的作用是通过一个数组创建一系列的标量变量，这些变量的名称必须是数

组中关键字的名称，而变量值则是数组中的值。

```
$array=array("key1"=>"value1","key2"=>"value2","key3"=>"value3");  
extract($array);  
echo $key1.$key2.$key3;
```

感谢收看本次教程！

本课程是由北风网(ibeifeng.com)

瓢城**Web**

俱乐部(yc60.com)

联合提供：

本次主讲老师：李炎恢

我的博客：hi.baidu.com/

李炎恢/

我的邮件：yc60.com@gmail.com