

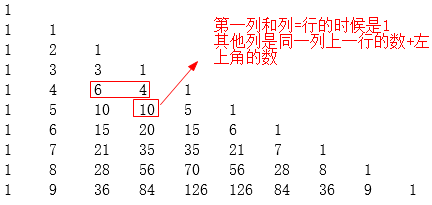
主讲教师：传智.神龙教主

传智播客 PHP学院 <http://php.itcast.cn> 发布

## 作业

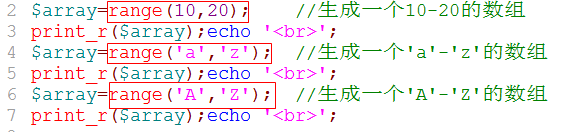
### 打印10行杨辉三角

1. <style type="text/css">
2. span{
3. width:45px;
4. height:20px;
5. display:inline-block;
6. }
7. </style>
8. <?php
9. //打印杨辉三角
10. function yanhui($lines=10) {
11. for($rows=1; $rows<=$lines; $rows++) {
12. for($cols=1; $cols<=$rows; $cols++) {
13. if($cols==1 || $cols==$rows)
14. $num[$rows][$cols]=1;
15. else {
16. $num[$rows][$cols]=$num[$rows-1][$cols]+$num[$rows-1][$cols-1];
17. }
18. echo '<span>'.$num[$rows][$cols].'</span>';
19. }
20. echo '<br>';
21. }
22. }
23. yanhui();

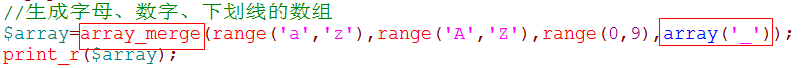


## 操作数组的函数

### range(起始值,结束值)：生成一个指定范围的数组



### array\_merge()数组合并



脚下留心：数组合并时候下标冲突，按如下规则解决：

1. 如果是字符串下标，后面的元素将前面的元素覆盖
2. 如果在整数下标，重新建立下标索引

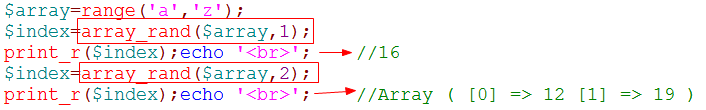


### array\_rand(数组，随机取出元素的个数)

随机取出数组内元素，返回随机数组的下标

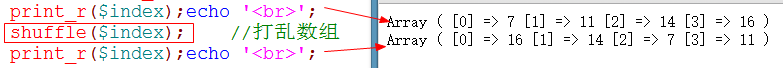
如果随机取出1个，返回整型下标

如果随机取出多个，返回整型下标数组

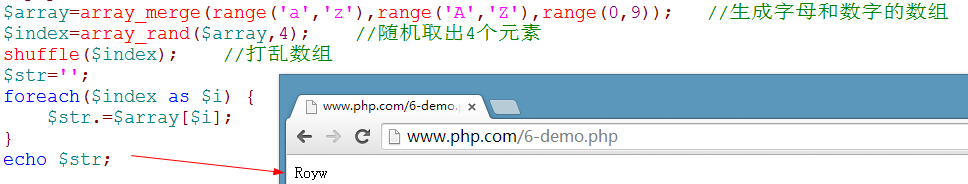


### shuffle(数组)：打乱数组

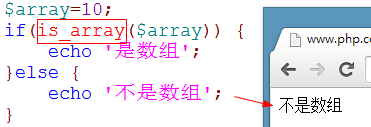
打乱成功返回true，打乱失败返回false。



### 例题：生成验证码字符串

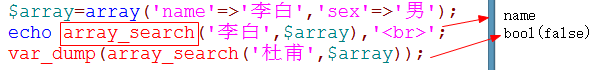


### is\_array():判断变量是否是数组类型



### array\_search()

在数组中搜索给定的值，如果成功则返回相应的键名，如果找不到返回false。



## 操作数组键值函数

### array\_keys()和array\_values()

1、array\_keys($array):返回数组中的所有键名

2、array\_values($array):返回数组中的所有值

$stu=array('name'=>'李白','sex'=>'男','age'=>20);

$keys=array\_keys($stu); //获取所有的键

$values=array\_values($stu); //获取所有的值

print\_r($keys);

echo '<br>';

print\_r($values);



### array\_combine(键数组,值数组)

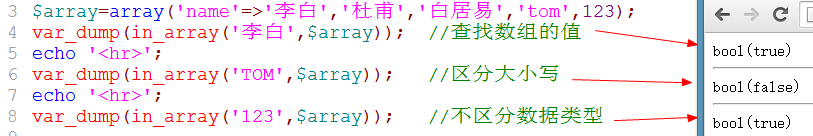
作用：创建一个数组，用键数组的值做键名，值数组的值做值。



脚下留心：如果键数组和值数组的元素个数不一样，返回false。

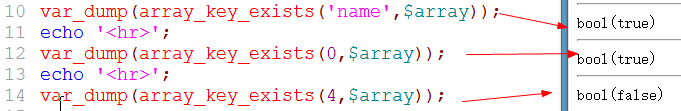
### in\_array(查找的值,数组)

作用：检查数组中是否包含某个值，返回bool类型



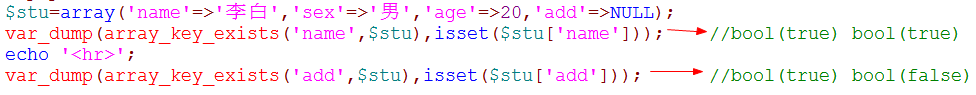
### array\_key\_exists(键,数组)；

检查键名或索引是否在数组中



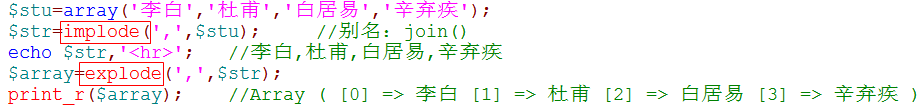
### isset(变量名)：检测变量是否设置，并且设置的值不为null

脚下留心：array\_key\_exists()和isset()的区别



## 数组连接和字符串分割函数

1. implode()：别名是join()，将数组的值连接成字符串
2. explode()：将字符串分割成数组

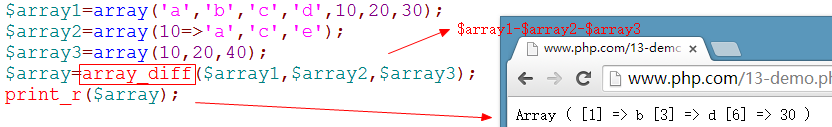


## 数组运算函数

### 数组合并：array\_merge(数组1,数组2,...)

参见前面的例题

### 数据的差集：array\_diff(数组1，数组2，数组3,…)



### 数组相交：array\_intersect(数组1,数组2,数组3,…)



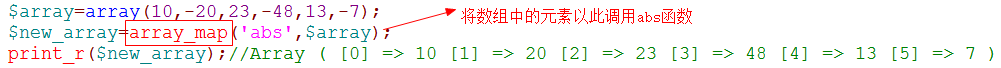
脚下留心：数组的交集结果中，下标使用的是第一个数组的下标。

## 使用回调函数的函数

回调函数：计算机自己调用的函数，不需要手动调用。

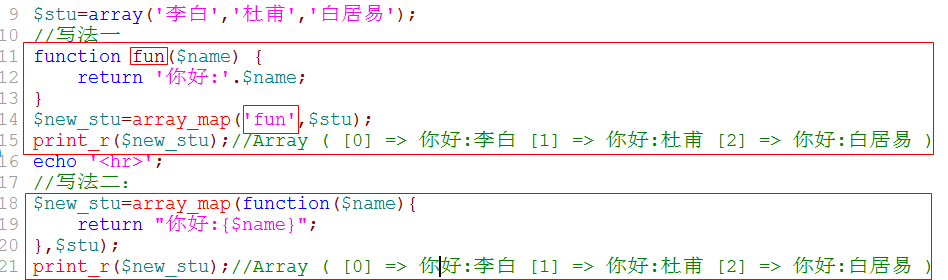
### array\_map('函数名','数组')

1、调用PHP自身的函数



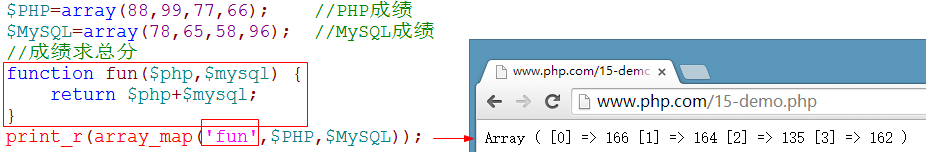
2、调用PHP自定义函数

在学生前面加上“你好：”

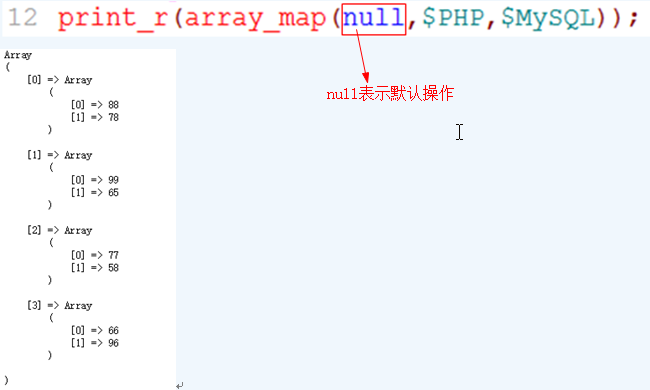


array\_map()支持一次处理多个数组，将所有数组中同位置的元素作为参数传递给回调函数。

求学生的总分



多学一招：array\_map()默认可以将多个数组的同位置的元素组成新数组。

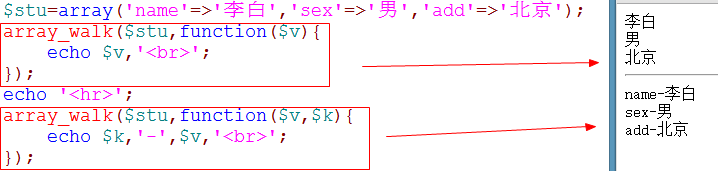


脚下留心：array\_map()只能处理值，不能处理键

### array\_walk(数组,回调函数)

作用：对数组中的每个元素的键值进行对调处理，类似于foreach循环

1. 如果回调函数是一个参数，则这个参数是值
2. 如果调用函数是两个参数，第一个是值，第二个是键

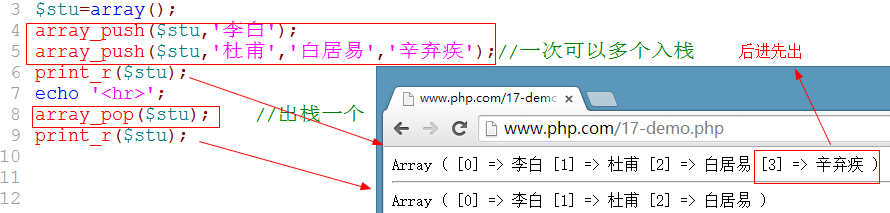


## 栈和队列

1. 栈和队列都属于数据结构
2. 栈是先进后出
3. 队列是先进先出

### 实现栈效果的函数

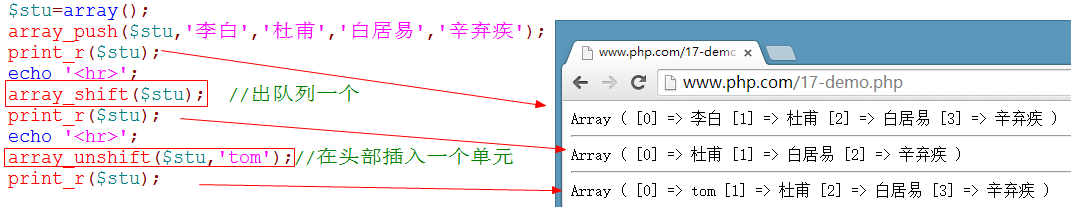
1. array\_push(数组,入[栈|队列]的值)
2. array\_pop()：出栈，将最尾部的元素弹出



注意：array\_push()和array\_pop()作用在于维护一个连续的存储空间；为新增加的元素重写下标（索引）。

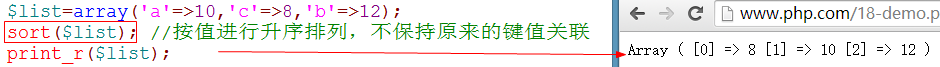
### 实现队列效果的函数

1. array\_shift()：出队列，将队列中的第一个元素移除
2. array\_unshift()：在数组的开头插入一个元素



## 数组排序函数

### sort() ：按值进行升序排列，不保持原来的键值关联



### rsort()：按值进行降序排列，不保持原来的键值关联



### asort()：按值进行升序排列，保持原来的键值关联



### arsort()：按值进行降序排列，保持原来的键值关联



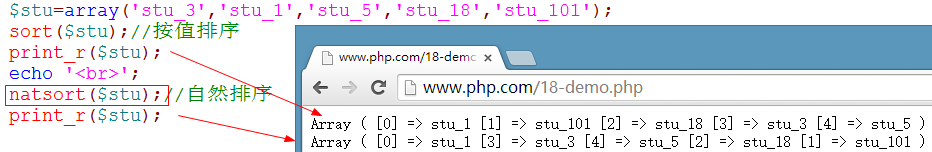
### ksort()：按键进行升序排列



### krsort()：按键进行降序排列



### natsort()：自然排序



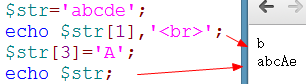
### usort()：用户自定义排序



## 字符串操作函数

### []

字符串可以理解成字符的集合，所以可以通过[]来访问。不能操作中文。



### strlen()

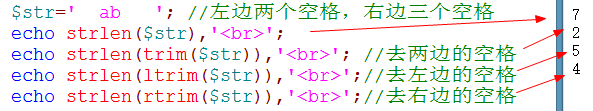
返回字符串长度，以字节为单位



脚下留心：在gbk下一个汉字2个字节，在utf8下一个汉字3个字节。

### trim()、ltrim()、rtrim()：去空白字符

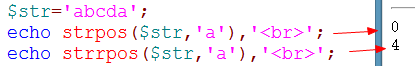
1. trim()：去字符串两边的把空格
2. ltrim()：去字符串左边的空格
3. rtrim()：去字符串右边的空格



### strpos()

strpos()：查找字符串首次出现的位置

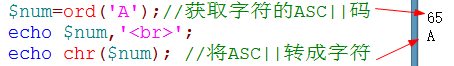
strrpos()：查找字符串最后一次出现的位置



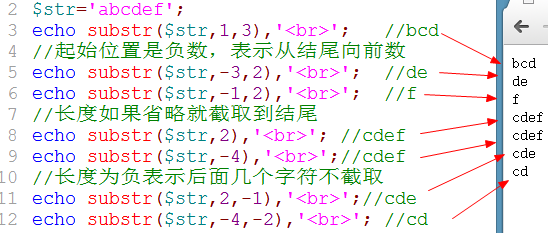
### str\_repeat()：重复字符串



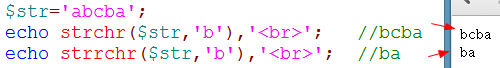
### chr()、ord()



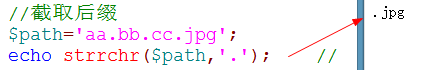
### substr(字符串,开始位置,截取长度):截取字符串



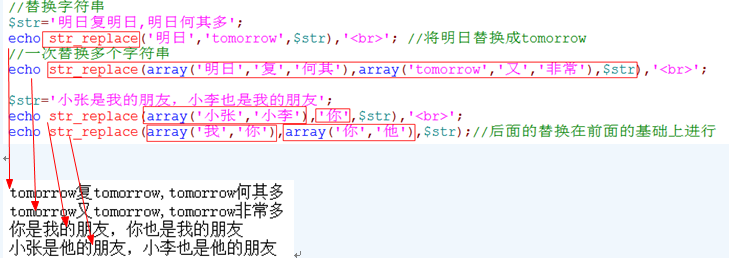
### strchr()、strrchr()



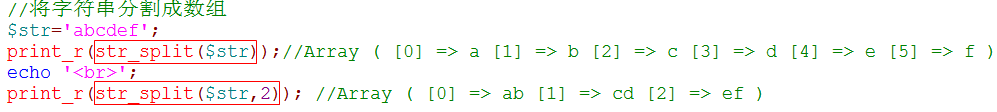
应用：截取文件的后缀



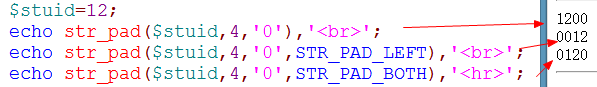
### str\_replace():替换字符串



### str\_split():将字符串分割成数组



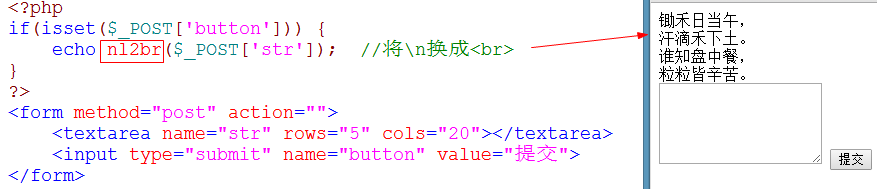
### str\_pad(待填充字符串,总位数,填充字符,填充位置)：填充字符串



### 字符串大小写转换



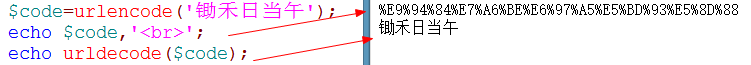
### nl2br()：将字符串的换行转成<br>



### htmlspecialchars()：将HTML字体转成实体



### urlencode()、urldecode()：url编码和解码



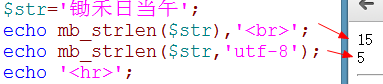
## 多字节字符串处理函数

PHP默认本身不支持多字节处理。要开启mb\_扩展（multibytes）.

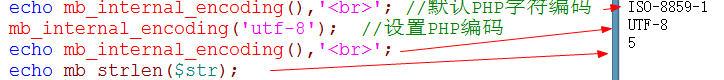


以mb\_开头的函数都在多字节处理函数。

### mb\_strlen()：获取字符串的长度



### mb\_internal\_encoding()：设置/获取内部字符编码



### mb\_strpos()：查找字符串在另一个字符串中的位置



### mb\_substr()：获取子字符串



## 字符串编码

### iconv(输入的编码，输出编码，字符串)

