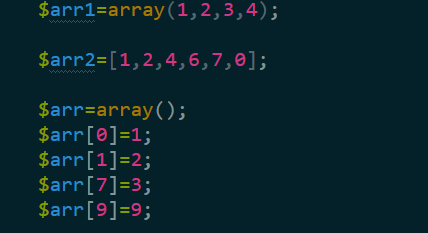
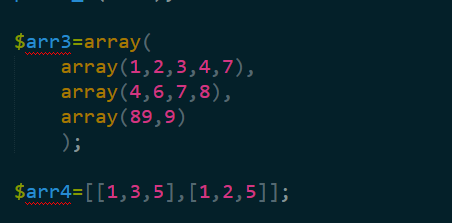
## <<--php的数组真尼玛恶心-->>

## 1.索引数组：

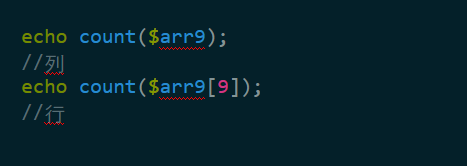


索引数组的下标可以不连续。

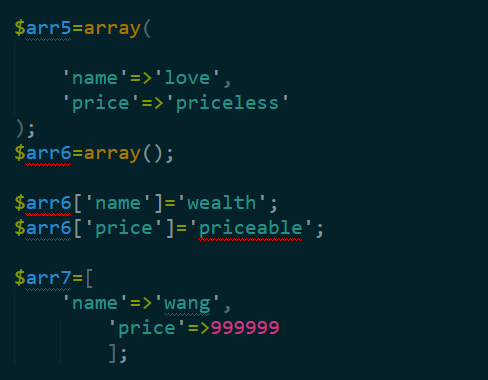
## 2.二维索引数组：



## 3.输出数组长度

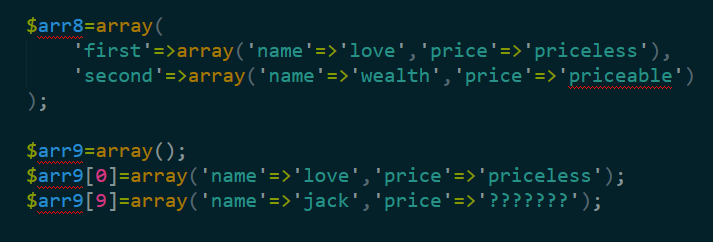


## 关联数组



## 4.二维关联数组：

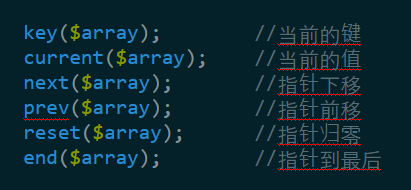
关联关联数组，索引关联数组：



## 数组遍历：

Php的数组一般是关联数组或者是不连续的索引数组，所以数组遍历是引入了指针的概念。

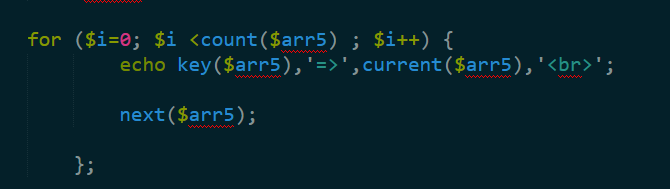
## .数组指针函数

* + 1. 
    2. 使用指针操作数组时应定要记得归位，（不会自动归位）。

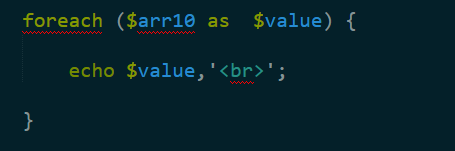
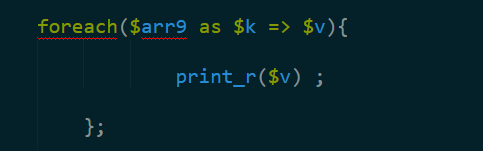
## .php数组特点

* + 1. Php数组的最后一位是布尔值，可以用来做判断，

## .指针数组遍历(自创)(foreach循环的原理)

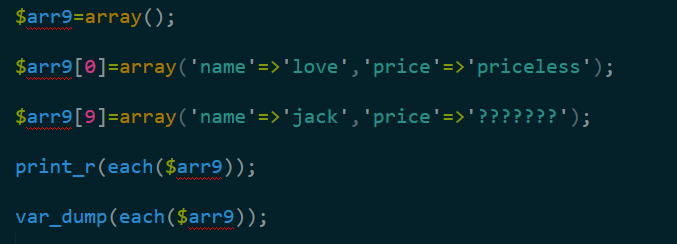
* + 1. 

## .foreach循环(键值可省略)

* + 1. 

## Each

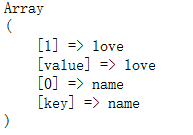
* + 1. 概念：each是一个非常恶心的东西（误！），他会把你当前数组指针位置的键和值按索引和关联的两种方式用一个数组展现出来，如果这是一个二维数组值对应值就是数组的展现相似。
    2. 二维的例子：



每一次each操作都会移动指针，所以别忘了那一套归位。

## 效果

## .list

这是一维数组使用了each之后的效果，之后我们可以使用

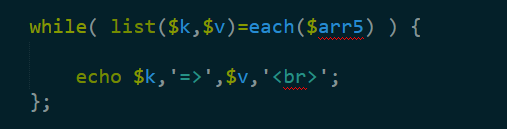
list对他进行列表转换，因为each会为我们转换一种索引数

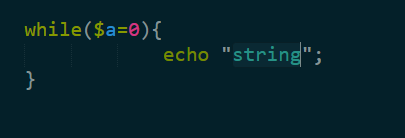
组的方式显示键值，而list只能对索引数组进行操作，所二 者连用。



这里的$k和$v就是name和love

## .while和list和each三者连用

1. .这里把 list列表化，因为赋值语句也可以做判断，前面的数据类型转换，只要部位空就都是真。

这个就不行是false，这里的单等号就是赋值，

之后到数组指针最后一位时each的值是false给$v判断循环体结束，

## .foreach和list连用用来解决索引二维数组。



不解释了，二维数组里的一维数组只能是索引数组。