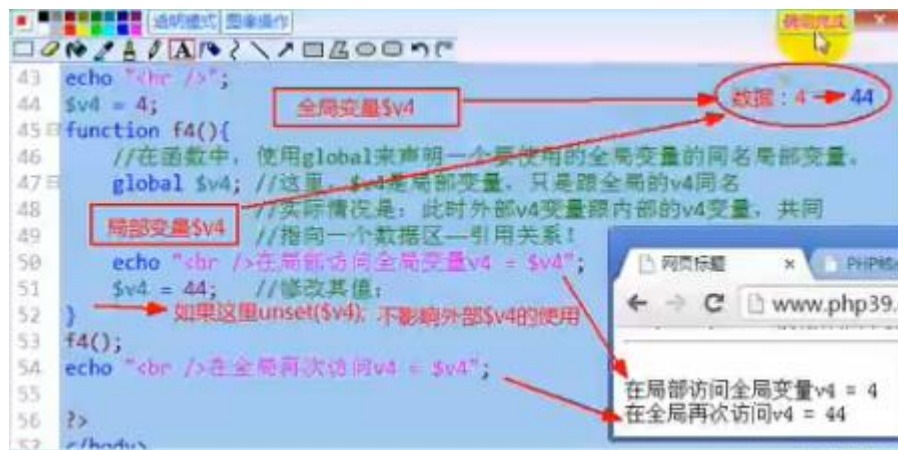


在局部作用域中引用全局变量

做法 1:



做法 2:



有关函数的系统函数:

- Function\_exists() : 判断一个函数是否被定义过,其中使用的参数为"函数名";
- Func\_get\_arg()
- Func\_get\_args()
- Func\_num\_args()


有关函数的编程思想

递归思想---递归函数

递归函数 : 就是在一个函数内部调用它自己的函数.

实用的递归函数是 : 能够控制这个调用的过程中,会在某个时刻(条件下)停下来.

```
14 //现在，假设，有一个函数，该函数“能够”计算n的阶乘。
15 function jiecheng( $n ){
16     echo "<br />开始：有人要求($n)的阶乘";
17     if( $n == 1){
18         echo "<br />结束：终于求到了($n)的阶乘： 1";
19         return 1;
20     }
21     $jieguo = $n * jiecheng($n-1);
22     echo "<br />结束：终于求到了($n)的阶乘： : $jieguo";
23     return $jieguo;
24 }
25 $v2= jiecheng(5); //结果应该是5的阶乘
26 /*
27 演示调用过程：
28 $v2 = jiecheng(5)相当于：
29 $v2 = 5 * jiecheng(4)==>>
30 $v2 = 5 * (4 * jiecheng(3) ) ==>>
31 $v2 = 5 * (4 * (3 * jiecheng(2) ) ) ==>>
32 $v2 = 5 * (4 * (3 * (2 * jiecheng(1) ) ) ) ==>>
33 $v2 = 5 * (4 * (3 * (2 * 1 ) ) ) ==>>
34 $v2 = 5 * (4 * (3 * 2 ) ) ==>>
35 $v2 = 5 * (4 * 6 ) ==>>
36 $v2 = 5 * 24 ==>>
37 $v2 = 120
38 */
39 echo "<br />$v2 = $v2";
40
```



## 递归思想的总结

当面对一个“大问题”,该大问题可以经由该问题的同类问题的“小一级问题”而经过简单计算获得;