

# 变量作用域

## 1. 变量作用域:

变量的作用域，就是指：一个变量，在什么范围中可以被使用的情况。

Php中，有3中变量作用域：

局部作用域：

指一个函数的内部范围

对应这样的变量，就称为“局部变量”

超全局作用域：

指所有的代码范围

对应这样的变量，就称为“超全局变量”

其实就只有几个：\$\_GET, \$\_POST, \$\_SERVER, \$\_REQUEST, \$GLOBALS

\$\_COOKIE, \$\_FILES, \$\_SESSION

全局作用域：

不在函数内部的范围——函数外部

对应这样的变量，就称为“全局变量”

通常：

### 1. 全局范围不能访问局部变量；

```
$v1 = 1;
function f1(){
    echo "<br />在函数内部访问外部: v1 = $v1;";
}
f1();
```

Notice: Undefined variable: v1 in  
在函数内部访问外部: v1 = ;

### 2. 局部范围不能访问全局变量；

```
//这里演示全局访问局部变量，出错:
function f2(){
    $v2 = 1;
}
f2();
echo "<br />在函数外部访问局部变量: v2 = $v2;";
```

Notice: Undefined variable: v2  
在函数外部访问局部变量: v2 = ;

3. 函数内部的变量(局部变量)，通常在函数调用执行结束后，就被“销毁”了。

4. 有一种局部变量，在函数调用结束后不会被销毁，它叫做“静态变量”

```
//静态变量
function f3(){
    static $c = 0; //此行赋值只走一次，每次进函数都取上一次的值
    $c++;
    $dd = 0;
    $dd++;
    echo "<br>c=$c, dd=$dd, 此函数被调用次数为$c";
}

f3();
f3();
f3();
```

c=1, dd=1, 此函数被调用次数为1  
c=2, dd=1, 此函数被调用次数为2  
c=3, dd=1, 此函数被调用次数为3

可见，静态变量可以用于统计一个函数调用次数，当然，也可以用于其他需要进行数据保留的函数调用过程。

## 2. 局部作用域使用全局变量

通常有两种做法：

```
//局部访问全局作用域
//做法1
$v4 = 1234;
function f4(){
    //方式一：使用global来声明一个要使用的全局变量的同名局部变量
    global $v4; //这里$v4是局部变量，只是和全局的v4同名
                //实际情况是此时外部的v4全局变量跟内部的局部v4变量共同指向一个数据区域引用关系
                //值修改了后会影响到全局的v4值
    echo "方式1:$v4<br>";

    //方式二：使用$GLOBALS[""]预定义超全局变量来读写访问全局作用域
    echo "方式2:". $GLOBALS["v4"]. "<br>";
}

f4();
```

方式1:1234

方式2:1234

注意，做法2一旦使用unset(\$超全局变量GLOBALS[“V4”])，那么该超全局变量就被释放了。而做法1，相当于存在两个引用关系，如果unset global声明的变量，相应全局的变量并不会被释放掉