## 函数中变量的作用域

### 案例描述

1. 考核知识点

函数中变量的作用域

1. 练习目标

* 掌握全局变量在函数内使用的方法

1. 需求分析

全局变量是不能直接在函数内部使用的，需要使用global关键字，或使用$GLOBALS超全局变量。假设在函数外定义一个$radius变量，然后在getArea()函数中直接使用$radius，与在函数getArea()中使用关键字global声明$radius，或是使用$GLOBALS进行对比。

1. 设计思路（实现原理）
2. 编写PHP文件，获取圆的半径$radius。
3. 编写getArea()函数，直接使用半径$radius，计算圆的面积。
4. 在函数中使用global关键字声明半径$radius，计算圆的面积。
5. 使用$GLOBALS获取半径$radius，计算圆的面积。
6. 对比输出结果。

### 案例实现

1. 编写PHP文件：index.php，完成HTML页面的设计。代码如下：

<!doctype html>

<html>

<head>

<meta charset="UTF-8">

<title>函数中变量的作用域</title>

<style>

h1{font-size:18px;text-align:center}

.box{width:350px;border:2px dotted #FE0D0B;padding:0 10px 10px 10px;margin:20px auto}

</style>

</head>

<body>

<div class="box">

<h1>函数中变量的作用域</h1>

<?php

// 此处编写PHP代码

?>

</div>

</body>

</html>

1. 继续编辑PHP文件，用于直接使用半径$radius，计算圆的面积。

<?php

function getArea()

{

$area = round(pi() \* pow($radius, 2), 2);

return $area;

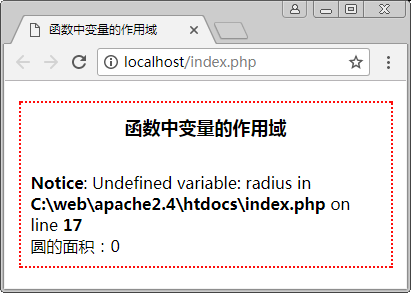
}

$radius = 4; // 圆的半径

echo '圆的面积：' . getArea();

?>

运行程序，结果如下图所示：



1. 修改getArea()函数，在函数中使用global关键字声明$radius。代码如下：

function getArea()

{

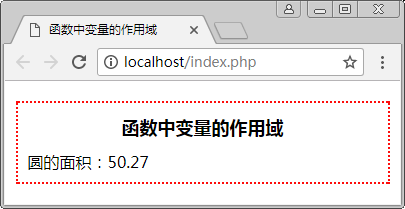
global $radius;

$area = round(pi() \* pow($radius, 2), 2);

return $area;

}

运行程序，结果如下图所示：



1. 修改getArea()函数，使用$GLOBALS获取$radius。代码如下：

function getArea()

{

$radius = $GLOBALS['radius'];

$area = round(pi() \* pow($radius, 2), 2);

return $area;

}

运行程序，运行结果与上一步相同。

### 案例总结

在函数内部使用外部变量时，可以通过如下3种方式实现：

1. 通过参数传递，传值或传引用。
2. 通过global关键字来提升。
3. 通过$GLOBALS来获取。