



## 6. PHP 数据类型

String（字符串），Integer（整型），Float（浮点型），Boolean（布尔型），Array（数组），Object（对象），NULL（空值）。

### 6.1 PHP字符串

一个字符串是一串字符的序列，就像 "Hello world!"。

你可以将任何文本放在单引号和双引号中：

```
<?php
$x = "Hello world!";
echo $x;
echo "<br>";
$x = 'Hello world!';
echo $x;
?>
```

### 6.2 PHP 整型

整数是一个没有小数的数字。

整数规则：

- 整数必须至少有一个数字 (0-9)
- 整数不能包含逗号或空格

- 整数是没有小数点的
- 整数可以是正数或负数
- 整型可以用三种格式来指定：十进制，十六进制（以 0x 为前缀）或八进制（前缀为 0）。

在以下实例中我们将测试不同的数字。

PHP `var_dump()` 函数返回变量的数据类型和值：

```
<?php
$x = 5985;
var_dump($x);
echo "<br>";
$x = -345; // 负数
var_dump($x);
echo "<br>";
$x = 0x8C; // 十六进制数
var_dump($x);
echo "<br>";
$x = 047; // 八进制数
var_dump($x);
?>
```

## 6.3 PHP 浮点型

浮点数是带小数部分的数字，或是指数形式。

在以下实例中我们将测试不同的数字。 PHP `var_dump()` 函数返回变量的数据类型和值：

```
<?php
$x = 10.365;
var_dump($x);
echo "<br>";
$x = 2.4e3;
var_dump($x);
echo "<br>";
$x = 8E-5;
var_dump($x);
?>
```

## 6.4 PHP 布尔型

布尔型可以是 TRUE 或 FALSE。

```
$x=true;
$y=false;
```

布尔型通常用于条件判断。在接下来的章节中你会学到更多关于条件控制的教程。

## 6.5 PHP数组

数组可以在一个变量中存储多个值。

在以下实例中创建了一个数组， 然后使用 PHP `var_dump()` 函数返回数组的数据类型和值：

```
<?php
$cars=array("Volvo","BMW","Toyota");
var_dump($cars);
?>
```

## 6.6 PHP 对象

对象数据类型也可以用于存储数据。

在 PHP 中，对象必须声明。

首先，你必须使用 `class` 关键字声明类对象。类是可以包含属性和方法的结构。

然后我们在类中定义数据类型，然后在实例化的类中使用数据类型：

```
<?php
class Car
{
    var $color;
    function __construct($color="green") {
        $this->color = $color;
    }
    function what_color() {
        return $this->color;
    }
}
?>
```

以上实例中PHP关键字 `this` 就是指向当前对象实例的指针，不指向任何其他对象或类。

## 6.7 PHP NULL值

NULL 值表示变量没有值。NULL 是数据类型为 NULL 的值。

NULL 值指明一个变量是否为空值。 同样可用于数据空值和NULL值的区别。

可以通过设置变量值为 NULL 来清空变量数据：

```
<?php
$x="Hello world!";
$x=null;
var_dump($x);
?>
```