

# 类型转换

## 1. 自动转换

在任何运算中，如果需要某种类型的数据，而给出的类型不是该类型，通常都会发生自动转换，将该类型转换为目标所需要的类型。

比如：`octdec($x), bindec($x)` //这里就要求\$*x* 是字符串，如果不是，就会自动转换。

比如：`$v1 = 1 + "2"` //此时也发生了自动转换

如果”*x*”中*x*为非数值，则取前面是数值部分，取不到则为0，如 `a12` 为0，`12a` 为12

## 2. 强制转换

自动类型转换是由”运算符”或类似运算符来决定的。

而

强制类型转换，仅仅是一个简单的语法：

形式：(目标类型)数据

含义：将该数据转换为设定的目标类型

例子：

```
1  <?
2      $v1 = "124";
3      var_dump($v1);
4
5  echo "<br>";
6
7      $v2 = (int)$v1;
8      var_dump($v2);
9
10 echo "<br>";
11
12     $v3 = (float)$v2;
13     var_dump($v3);
14 ?>
```

通常的转换目标类型有：

`(int)`, `(float)`, `(string)`, `(array)`, `(object)`

上述强制类型转换，并不改变该变量本身的数据或类型，如果要直接改变某个变量的数据或类型，则需调用以下语法：

`settype($变量名, 目标类型);`

### 3. 类型相关的系统函数

(1) `var_dump()`: 用于输出变量的“完整信息”，几乎只用于调试代码

(2) `getType($变量名)`: 获取该变量的类型名字，返回的是一个表示该类型的字符串

(3) `setType($变量名, 目标类型)`: 将该变量强制改为目标类型

(4) `isset()`、`empty()`、`unset()`

(5) `is_xx 类型()` 系列函数: 判断某个数据是否为某种类型，有如下类型:

`is_int($x);`

`is_float($x);`

`is_string($x);`

`is_bool($x);`

`is_numeric($x);` // 判断是否是一个数字

`is_scalar($x);` // 判断是否是标量类型

`is_array($x);`

`is_null($x);`

`is_object($x);`