

123  $\Rightarrow$  十进制

0123  $\Rightarrow$  八进制

0b11  $\Rightarrow$  二进制

0x1E  $\Rightarrow$  十六进制

其中 0b [0B] 和 0x [0X] 不区分大小写.

当某值超出精确表示范围后会被  
转换为 float, 即使看着一样, 实际上是  
取约数的结果

int div(a, b)  $a$  整除  $b$

$a \% b$   $a$  除以  $b$  取余 还是  
`round()` 四舍五入为整数 (float 型)

当字符串型与 int 比较或运算时要  
当心, php 中其他类型到整形的转换  
可能出现不可预计的结果.

如:  
 $0123 = 10123$  (八进制) ; // false.

自动转换是通过将 ' ' 中第一个非  
零数以前 (对整数) 和第一个字符以后  
的内容省略, 剩余当成 10 进制看的.

容

'01x23'  $\Rightarrow$  1

'x0123'  $\Rightarrow$  0 (无可选内容时为0)

'102x35'  $\Rightarrow$  102

除默认的 (int) '\_\_\_' 转换方法, 对于数字或被 ' ' 标记的正确格式数字, 可用 `intval(' ', 底)` 按对应底来认为, 并返回 int 值 (十进制)

如:

`intval('16', 7)  $\Rightarrow$  int(13)`

`intval('0x20', 16)  $\Rightarrow$  int(32)`