# 浮点类型

浮点数2种表示形式

常规写法$v1=123.456;

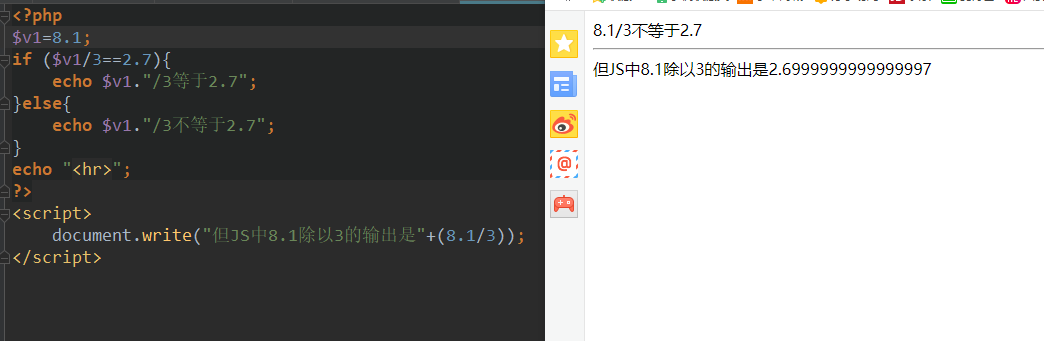
科学写法 $v1=123.456E2；含义：123.456乘以10的二次方

$v2=123.456E3； 含义：123.456乘以10的三次方

$v3=123E4；含义为：123乘以10的四次方，还是浮点数

**浮点数使用的细节知识**

浮点数不应进行大小比较



因为：逃不过二进制的缺陷 有它内部的机制 提前无效

1. 所有数字，最终表示形式，都是2进制！！
2. 大多数浮点数的2进制形式，不能完全表达准确！！最终只能以很高的精度接近理论值；
3. 因此，浮点数比较是不可靠的；

再从另一个角度证明浮点数的准确性；

说明：php中输出其实是做了一定的处理之后的显示结果；而js是真实反映；

那该怎么办？？

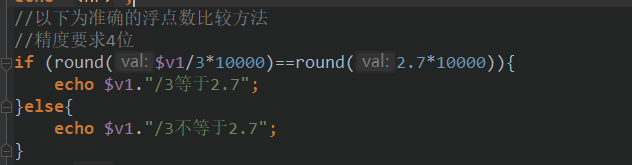
考虑实际应用所需的精度情况下。去将要比较的浮点数，转换为整数之后再比较

比如：

要求精度为3位小数，则都乘以1000，然后取整后比较

要求精度为4位小数，则都乘以10000，然后取整后比较

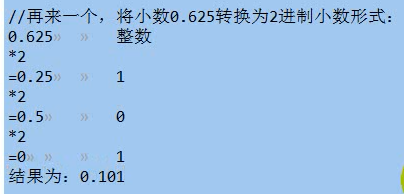
………..以此类推

# 小数转二进制的做法：乘2并顺序取整数部分



为什么32位？因为系统只能32位



当整数运算结果超出整数的范围后，会自动转换后浮点数

获取一个数据的类型函数有：

* GetTYPE（$变量名）
* Var\_dump($变量名)会输出该变量类型数据内容及长度

