1，布尔值 true，false 不区分大小写，且不是常量。

5 PHP 数据类型

1 整型：int integer　　PHP_INT_SIZE (字节数)　　PHP_INT_MAX(整型的最大值)

2 浮点型：float

3 布尔型： bool boolean ，　两个值：true　false 不区分大小写（true，false 不是常量）

当做 False 情况：1 布尔值 false 本身　　2 整型值 0　　3 浮点型 0.0

4 空字符串" 以及空字符串'0'

5 特殊数据类型 null（包括尚未赋值的变量--空数组）

2，单引号和双引号都可以定义字符串

单引号不解析包含的变量，而双引号会解析包含的变量；

输出单引号中包含的单引号用转义字符（反斜杠）；

输出单引号中的双引号不需要转义；

输出双引号中的双引号，用转义符（反斜杠）；

输出双引号中的单引号无需转义。

单引号与双引号定义字符串的区别：

1 单引号不解析其中的变量

2 最外层为双引号，里面包含单引号再包含变量，解析变量

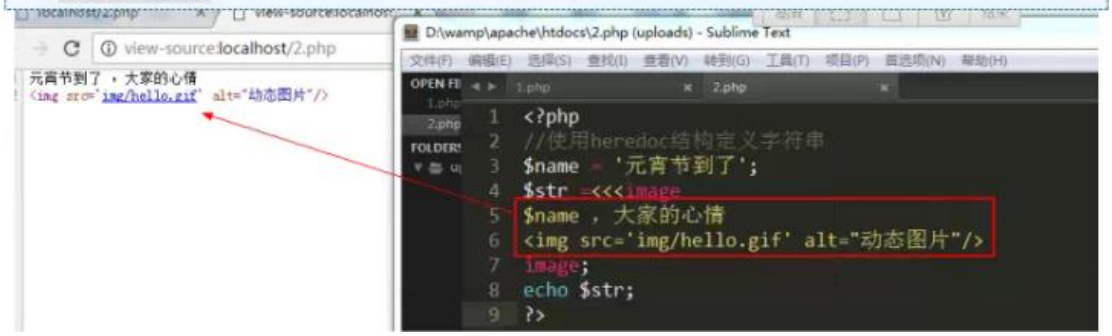
3 最外层为单引号，里面包含了双引号再包含变量，不解析变量

建议： 定义字符串变量尽量使用单引号，执行效率高。

3，Heredoc 结构定义字符串

单引号中包含单引号，双引号里面包含了双引号或者包含了变量，定义时需要使用转义符比较麻烦。使用 heredoc 结构定义字符串

```
$str =<<<heredoc 标记
 字符串值
Heredoc 标记;
```



使用 heredoc 注意：

1 解析其中变量

2 单引号双引号不需要转义

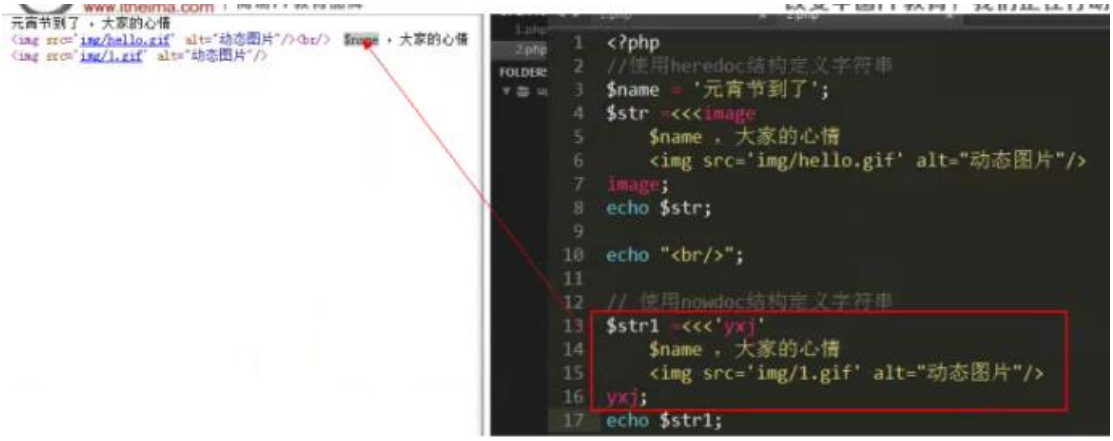
3 标记可以自定义，遵循 PHP 变量命名规则

4 开始标记与结束标记必须一样

5 结束标记必须顶格写（当前行的顶一个字符位置）

4，newdoc 结构定义字符串

```
$str =<<<'标记'
    字符串值
标记;
```



Nowdoc 结构：

1 开始标记与结束标记一致
2 开始标记使用单引号包括起来
3 结束标记必须顶格写
4 不解析其中变量
5 单引号双引号不需要转义

Heredoc 与 nowdoc 区别

Heredoc 解析变量，类似双引号
使用 nowdoc，开始标记使用单引号包括起来

Heredoc 与 nowdoc 相同点：

主要用于定义多行文本。

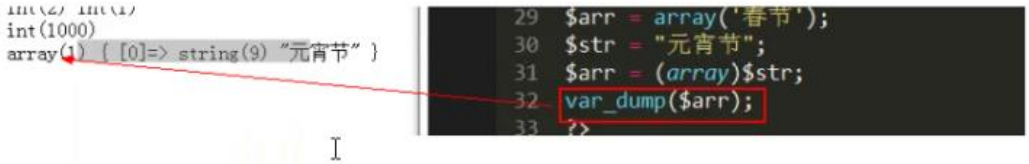
5，特殊数据类型

null：赋值为 null；unset；没有定义。

resource 资源类型，相当于第三方数据的应用，比如数据库。

1.9. 字符串强制转换为数组

数组已经定义，自动清空数组的元素再转换。



自动转换及强制转换比较

1 转换之后，原始数据类型不发生转变，只是参与运算的值发生改变。

2 自动转换（根据环境）的原则与强制转换（人为转换）的原则一致。

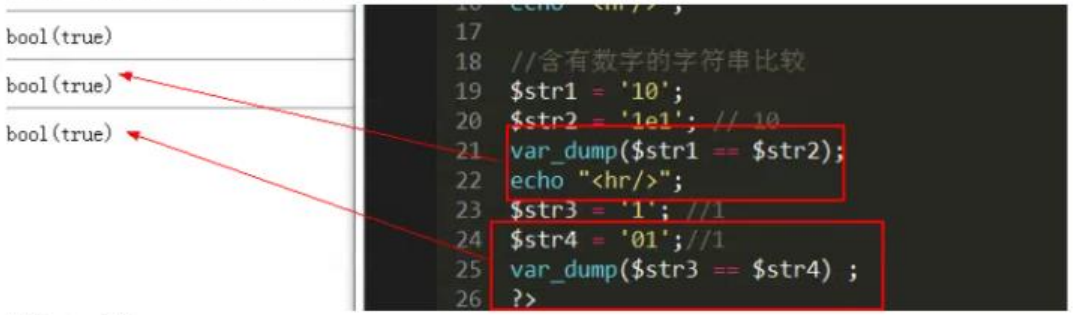
运算原则：

除全等及全不等之外的比较运算符的运算原则

不同类型之间比较，会出现类型的自动转换。

不同类型之间转换原则： 布尔类型》整型》字符串

含有数字的字符串之间比较，先转换为数字，在进行比较。



全等、全不等：

不进行类型的自动转换。即判断数值，又判断类型，只有两者均相等时，才全等。

1. 13. 逻辑运算符

与（&& and） 或（|| or） 非（！）

运算原则：

与： 全真为真，否则为假

或： 全假为假，否则为真

非： 非真即假，非假为真

```php
var_dump(true && true );  //true
var_dump(true && false); //false

var_dump(true || false); //true
var_dump(false || false ); //false

var_dump(!true); //false
var_dump(!false); //true
```

三大流程控制

1）顺序结构：从上往下依次执行；

2）分支（选择）结构：满足指定条件，执行指定代码；

3）循环结构：满足一个条件，循环执行指定的代码。