

复习

推荐使用标签

```
<php PHP 代码 ?>;
```

注释

单行注释 //

多行注释 /* */

变量的命名规则

建议使用小驼峰,命名要有意义

severName

常用输出语句

Echo

Print

Var_dump

Printf

PHP 的数据类型

整型 int integer // PHP_INT_SIZE(字节数) PHP_INT_MAX (整型的最大数);

浮点型 float

布尔型 bool Boolean

True && flase

定义字符串类型的变量

单引号定义字符串

单引号中包含了单引号,需要使用转义字符

单引号中包含了双引号,不需要去转义;

*单引号不识别\之外的转义;

双引号定义字符串

双引号中包含双引号使用\"

双引号包含单引号不需要转义

区别

单引号不解析其中的变量

最外层为双引号,内层单引号包含变量,会解析变量;

最外层为单引号,内层是双引号包含变量,不会解析变量;

*定义字符串时尽量使用单引号,可以提高效率;

Heredoc 结构

语法

```
$str = <<< heredoc //标记
```

需要定义的字符串

```
heredoc; // 要与标记一致,并需要顶格写
```

- 会解析其中变量
- 单引号双引号不需要转义

- 标记可以自定义,遵循 PHP 变量命名规则
- 开始标记与结束标记必须一致
- 结束标记必须顶格写 (当前行的顶一个字符位置)

Nowdoc 结构

```
$str = <<<'nowdoc'    //要加单引号括起来
    需要定义的字符串
nowdoc                //必须顶格写
```

- 开始标记与结束标记必须一致
- 开始标记使用单引号包括起来
- 结束标记必须顶格写
- 不解析其中变量
- 单引号双引号不需要转义

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

相同点

主要用于定义多行文本

特殊数据类型

Null 类型

比较运算符

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

不同类型之间的转换原则：布尔类型>整型>字符串

全等.全不等：

不进行类型的自动转换,既判断数值,又判断类型