

后盾网 人人做后盾

www.houdunwang.com

字符串

后盾网 2011-2013 v3.0

定义

- 字符串(string)是PHP中重要的数据类型之一。
- 字符串就是一系列字符。一个字符串可以只是一个字符，也可以非常巨大，由任意多个字符组成。

语法

- 字符串有三种定义方法：

单引号 (')

双引号 (")

定界符 (<<<)

字符串的定义

单引号

- 指定一个简单字符串的最简单的方法是用单引号（'）括起来。

例如：`$str1= 'houdunwang' ; //字符串str1赋值`

几点说明

- 1.输出单引号需要转义：`\'`
- 2.在'前输出反斜杠(`\`)，也需要转义：`\\`
- 3.单引号中的变量不会被解析

引号

双引号

- 双引号字符串最重要的一点是其中的变量名会被变量值替代,必要时需要加{}把变量名括起来。

几点说明

- 双引号里的变量可以直接被解析,有必要时要用{}括起来
- 转义字符:

转义符	含义
\"	双引号
\n	换行
\r	回车
\t	制表符
\\$	美元符
\\	反斜杠

转义

实例：

```
<?php
```

```
    $aa='aa';
```

```
    print('aa is \'aa\'.'" <br/>");
```

```
    print("aa is \"aa\""." <br/>");
```

```
    print("aa is 'aa'"." <br/>");
```

```
    print( "aa is $aa" . " <br>" );//变量被值替换
```

```
    print( "aa is {$aa}" . " <br>" );//大括号分离变量
```

```
    print('aa is $aa.'" <br>");
```

```
    print('aa is '.$aa);
```

```
?>
```

转义

定界符<<<

定界符文本表现的就和双引号字符串一样，在定界符文本中不需要转义引号。

注：结束符必须从第一列开始，并且后面除了分号以外不能包含任何其他字符，空格也不可以。

示例：

- `$str=<<<here`
- 这里放入内容即可
`here;`
- `echo $str;`

定界符

echo()

语法: echo "string arg1, string [argn]...";

它是语言结构，不是真正的函数。

它和print的区别在于，可以接受多个参数。

⑩ 例如：echo "first" ," second" ," third" ;//合法

⑩ echo ("hello" ," world!"); //解析错误

strlen()

取得字符串长度传度的函数

输出函数

函数:trim()

截去字符串首尾的内容

语法: string trim(string str);

返回值: 字符串

本函数返回字符串 string 首尾的空白字符去除后的字符串。

函数:ltrim()

语法: string ltrim(string str);

返回值: 字符串

本函数用来删去字符串中的前导空格 (whitespace)。

函数:rtrim()

语法: string rtrim(string str);

返回值: 字符串

本函数用来删去字符串中的后缀空格 (whitespace)。

字符串处理函数

函数: strtolower()

语法: string strtolower(string str);

返回值: 字符串

本函数将字符串 str 全部变小写字符串。

函数: strtoupper()

语法: string strtoupper(string str);

返回值: 字符串

本函数将字符串 str 全部变大写字符串。

字符串处理函数

函数:ucfirst()

将字符串第一个字符改大写。

语法: string ucfirst(string str);

返回值: 字符串

本函数返回字符串 str 第一个字的字首字母改成大写。

函数:ucwords()

将字符串每个字第一个字母改大写。

语法: string ucwords(string str);

返回值: 字符串

本函数返回字符串 str 每个字的字首字母全都改成大写。

字符串处理函数

函数:md5()

语法: string md5(string str);

返回值: 32字节字符串

本函数用来返回一个加密后的十六进制字符串

字符串处理函数

explode -- 使用一个字符串分割另一个字符串
返回由字符串被拆分后的的数组。

例如：

```
$pizza = "piece1,piece2,piece3";  
$pieces = explode( ",", $pizza);  
echo $pieces[0];    // piece1  
echo $pieces[1];    // piece2
```

字符串处理函数

implode()

将数组合并为字符串

语法:string implode (string glue, array pieces)

例如：

```
$array = array( 'lastname' , 'email' , 'phone' );
```

```
$comma_separated = implode( "," , $array);
```

```
echo $comma_separated; //
```

显示lastname,email,phone

字符串处理函数

函数:substr()

截取字符串。

语法: string substr(string string, int start, int [length]);

本函数将字符串 string 的第 start 位起的字符串取出 length 个字符，若省略参数 length，则取到字符串末尾。若 start 为负数，则从字符串尾端往前开始提取。如果length为整数，表示返回length个字符，若为负数，则表示取到倒数第 length 个字符。

示例

- echo substr('abcdef', 1, 3); // bcd

字符串处理函数

函数: `strchr()` 别名: `strstr()`

返回字符串中某字符串开始处至结束的字符串。

语法: `string strstr(string haystack, string needle);`

函数: `strrchr()`

返回字符串中某字符串开始处至结束的字符串。

语法: `string strrchr(string haystack, string needle);`

示例

- `$email = 'name@example.com';`
- `$domain = strstr($email, '@');`
- `echo $domain; // 打印 @example.com`

字符串处理函数

函数:strpos()

寻找字符串中某字符最先出现的位置。默认从 0 开始。

语法: `int strpos(string haystack, string needle, int [offset]);`

函数:strrpos()

寻找字符串中某字符最后出现的位置。

语法: `int strrpos(string haystack, char needle);`

示例

- `$pos = strpos('abc', 'a'); //输出0`

字符串处理函数

str_replace()

- 字符串替换，三种替换方式
- str_replace(string \$search, string \$replace, string \$str);
- str_replace(array \$search, string \$replace, string \$str);
- str_replace(array \$search, array \$replace, string \$str);

str_ireplace();//不区分大小区

字符串处理函数

URL编码处理

- //替换所有非字母数字的字符,变为%后面跟两位16进制,空格变为+
- urlencode();
- //对已%##编码的URL进行解析还原
- urldecode();

HTML实体的处理

- htmlspecialchars(); (把指定特殊符号转换成实体,如<>)

引号的处理

- addslashes(); (把' " \ NULL(\0) 4个字符前添加\显示)
- stripslashes(); (是addslashes的反函数,显示转移后的字符)

HTML标签的清理

- strip_tags(); (清理字符串中的html标签)
- 允许有第二个参数,为不去除哪个标签

将提交的\$_POST['message']进行实体与转义处理
创建一个检测文件扩展名是否为图片的函数

作业