**1, Nguyên tắc xử lý chuỗi.**

-Nguyên tắc đầu tiên phải kể đến ở đây là nếu bạn xử dụng dấu nháy "" để in ra chuỗi thì khi chèn biến bạn sẽ không phải nối chuỗi.

**VD**:

$variable = 'Toidicode.com';

echo "Website học lập trình online $variable";

// output: Website học lập trình online Toidicode.com

-Và nếu như khi sử dụng dấu nháy "" để in ra chuỗi mà trong chuỗi có dấu " thì bạn phải thêm ký tự \ trước nó.

**VD**:

echo "Anh ấy nói \" Toidicode.com dạy lập trình online!\"";

-Và cũng tương tự với dấu '

**VD**:

echo "Anh ấy nói \' Toidicode.com dạy lập trình online!\'";

**2, Các hàm xử lý chuỗi thông dụng.**

**addcslashes($str, $char\_list)**

-Hàm này có tác dụng chèn \ vào trước các ký tự trong chuỗi $str với các ký tự được liệt kê ở $char\_list

**VD**:

echo addcslashes("Toidicode.com", 'o');

// output: T\oidic\ode.c\om

echo addcslashes("Toidicode.com", 'a..z');

//output: T\o\i\d\i\c\o\d\e.\c\o\m

echo addcslashes("Toidicode.com", 'a..zA..Z');

// output: \T\o\i\d\i\c\o\d\e.\c\o\m

**addslashes($str)**

-Hàm này sẽ thêm ký tự \ vào trước các ký tự ', ", \ trong chuỗi $str nếu có.

**VD**:

echo addslashes("To'id'icod'e.com");

//output: To\'id\'icod\'e.com

**bin2hex($str)**

-Hàm này có tác dụng chuyển đổi chuỗi về dạng ASCII HEX của từng ký tự trong chuỗi $str.

**VD**:

echo bin2hex("Toidicode");

//ouutput: 546f696469636f6465

**chop($string, $charList)**

-Hàm này có tác dụng xóa ký tự,hoặc từ cuối cùng của chuỗi nếu nó = $charlist.

**VD**:

echo chop("Vũ Thanh Tài Thanh", "Thanh");

//output: Vũ Thanh Tài

**crc32($string)**

-Hàm này có tác dụng chuyển một chuỗi thành một số nguyên.

**VD**:

echo crc32("Toidicode.com");

//output: -1511835000

**explode($separator, $string, $limit)**

-Hàm này có tác dụng tách chuỗi $string thành nhiều chuỗi khác với điều kiện $separator, và giới hạn $limit.

**VD**:

$data = explode('o', "Toidicode.com");

print\_r($data);

//output: Array ( [0] => T [1] => idic [2] => de.c [3] => m )

$data = explode('o', "Toidicode.com", 2);

print\_r($data);

//output: Array ( [0] => T [1] => idicode.com )

**implode($separator, $array)**

-Hàm này có tác dụng nối tất cả các phần tử của mảng $array thành chuỗi với khoảng phân biệt $separator.

VD:

$data = [

'Toi',

'D',

'i',

'co',

'de',

'.com'

];

echo implode($data) . "<br>";

//output: Tidicode.com

echo implode('-', $data);

**strlen($string)**

-Hàm này có tác dụng đếm xem chuỗi $string có bao nhiêu ký tự.

**VD**:

$data = "toidicode.com";

echo strlen($data);

//output: 13

**str\_word\_count($string)**

-Hàm này có tác dụng đếm xem chuỗi $string có bao nhiêu từ.

**VD**:

$data = "toidicode.com";

echo str\_word\_count($data);

//output: 2

**str\_repeat($string, $repeat)**

-Hàm này có tác dụng lặp chuỗi $string $repeate lần.

**VD**:

$data = "toidicode.com";

echo str\_repeat($data,4);

//output: toidicode.comtoidicode.comtoidicode.comtoidicode.com

**str\_replace($find, $replace, $string)**

-Hàm này có tác dụng tìm kiếm chuỗi $find và thay thế chuỗi đó bằng $replace trong chuỗi $string.

**VD**:

$data = "toidicode.com";

echo str\_replace("t","T",$data);

//output: Toidicode.com

**md5($string).**

-Hàm này có tác dụng chuyển chuỗi $string sang mã hóa md5.

**VD**:

$data = "toidicode.com";

echo md5($data);

//output: d04742d12d431f009e90ef4239e6a782

**sha1($string)**

-Hàm này có tác dụng mã hóa chuỗi $string sang sha1.

**VD**:

$data = "toidicode.com";

echo sha1($data);

//output: 49debc628ec279341836be4485897afd5af078ae

**htmlentities($string), htmlspecialchars($string)**

-Hàm này có tác dụng chuyển đổi dữ liệu sang dạng thô.

**VD**:

$data = "<b>toidicode.com</b>";

echo htmlentities($data);

//output: <b>toidicode.com</b>

$data = "<b>toidicode.com</b>";

echo htmlspecialchars($data);

//output: <b>toidicode.com</b>

**htmlspecialchars\_decode($string), html\_entity\_decode($string)**

- Hàm này có tác dụng chuyển đổi dữ liệu về dạng thực của nó. các bạn có thể thử lại với ví dụ trên.

**strip\_tags($string, $allow)**

- Hàm này có tác dụng loại bỏ các thẻ html có trong chuỗi $tring, trừ các thẻ được cho phép $allow.

**VD**:

$data = "<b>toidicode.com</b>";

echo strip\_tags($data);

// output: toidicode.com

echo strip\_tags($data, '<b>');

//output: <b>toidicode.com</b>

**substr(string,start,length)**

-Hàm này có tác dụng cắt chuỗi $string, bắt đầu ở vị trí $strat và có giới hạn $length.

**VD**:

<?php

<?

php

// Positive numbers:

echo substr("Hello world", 10) . "<br>"; //d

echo substr("Hello world", -1) . "<br>"; //d

echo substr("Hello world", 0, 10) . "<br>"; //Hello worl

echo substr("Hello world", 0, -1) . "<br>"; //Hello worl

?>

**strtolower($string)**

-Hàm có tác dụng chuyển chuỗi $string sang in thường.

**VD**:

echo strtolower("TOIDICODE.COM");

//output: toidicode.com

**strtoupper($string)**

-Hàm có tác dụng chuyển đổi chuỗi $string sang hin hoa.

**VD**:

echo strtoupper("toidicode.com");

//output: TOIDICODE.COM

**ucword($string)**

-Hàm có tác dụng chuyển đổi chữ cái đầu tiên của các từ trong chuỗi sang in hoa.

**VD**:

echo ucwords("vũ thanh tài");

//output: Vũ Thanh Tài

**ucfirst($string)**

-Hàm có tác dụng chuyển đổi chữ cái đầu tiên trong chuỗi sang in hoa.

**VD**:

echo ucfirst("vũ thanh tài");

//output: Vũ thanh tài

**trim($string, $charlist)**

-Hàm có tác dụng loại bỏ khoảng trắng hoặc các ký tự $charlist trong chuỗi $string.

**VD**:

echo trim(" vũ thanh tài ");

//output: Vũ thanh tài

echo trim("Vvũ Thanh Tài", 'V');

//output: vũ Thanh Tài

**ltrim($tring,$charlist)**

-Tương tự như hàm trim nhưng chỉ cắt bên trái của chuỗi.

**rtrim($tring,$charlist)**

-Tương tự như hàm trim nhưng chỉ cắt bên phải của chuỗi.