

FUNCIONES STRING PHP

Estas son las funciones string, con su respectiva descripción.

FUNCIONES STRING EN PHP

| | | |
|--|---|--|
| <ul style="list-style-type: none">• FUNCIÓN STR_REPLACE | <p>Bienvenidos a la página</p> <p>↓</p> <p>Ingresen a la página</p> | <p>Reemplaza caracteres dentro de una cadena de caracteres</p> <pre>str_replace("cadena a buscar", "cadena de reemplazo", \$variableOFraseOriginal)</pre> <pre>echo str_replace("Bienvenidos","Ingresen",\$texto);</pre> |
| <ul style="list-style-type: none">• FUNCIÓN STRTOLOWER | <p>CADena de teXTO</p> <p>↓</p> <p>cadena de texto</p> | <p>Convierte todo a minúsculas</p> <pre>echo strtolower(\$texto);</pre> |
| <ul style="list-style-type: none">• FUNCIÓN STRTOUPPER | <p>CADena de teXTO</p> <p>↓</p> <p>CADENA DE TEXTO</p> | <p>Convierte todo a mayúsculas</p> <pre>echo strtoupper(\$texto);</pre> |
| <ul style="list-style-type: none">• FUNCIÓN COUNT_CHARS | <p>La planta azul</p> <p>↓</p> <p>N. veces 2 caracter " " en cadena N. veces 1 caracter "L" en cadena N. veces 4 caracter "a" en cadena N. veces 2 caracter "l" en cadena N. veces 1 caracter "n" en cadena N. veces 1 caracter "p" en cadena N. veces 1 caracter "t" en cadena N. veces 1 caracter "u" en cadena N. veces 1 caracter "z" en cadena</p> | <p>Conteo de caracteres</p> <pre>foreach (count_chars(\$data, 1) as \$i => \$val)</pre> <pre>echo "N. veces \$val caracter \"", chr(\$i) , "\" en cadena\n";</pre> |
| <ul style="list-style-type: none">• FUNCIÓN SUBSTR_COUNT | <p>--> Lunes Martes Lunes Jueves Lunes Viernes</p> <p>--> Lunes</p> <p>↓</p> <p>Lunes aparece (3) veces</p> | <p>Número de veces que aparece una cadena</p> <pre>echo " \$lista2 aparece (" .substr_count(\$lista1,\$lista2).") veces";</pre> |
| <ul style="list-style-type: none">• FUNCIÓN SUBSTR_COUNT | <p>--> Lunes Martes Lunes Jueves Lunes Viernes</p> <p>--> Lunes</p> <p>↓</p> <p>Lunes aparece (3) veces</p> | <p>Número de veces que aparece una cadena</p> <pre>echo " \$lista2 aparece (" .substr_count(\$lista1,\$lista2).") veces";</pre> |
| <ul style="list-style-type: none">• FUNCIÓN STRLEN | <p>La botella es blancaLa botella es blanca</p> <p>↓</p> <p>Longitud : 20</p> | <p>Longitud de una cadena</p> <pre>echo "Longitud : ".strlen(\$texto).";</pre> |
| <ul style="list-style-type: none">• FUNCIÓN TRIM, LTRIM, RTRIM, CHOP | <p>****BIENVENIDO****</p> <p>↓</p> <p>BIENVENIDO</p> | <p>Elimina espacios en blanco, saltos, caracteres, etc.</p> <pre>echo trim(\$texto,"*");</pre> |
| <ul style="list-style-type: none">• FUNCIÓN STR_REPEAT | <p>planta</p> <p>↓</p> <p>planta planta planta planta</p> | <p>Repite la cadena n veces</p> <pre>echo str_repeat(\$texto,4);</pre> |
| <ul style="list-style-type: none">• FUNCIÓN STRSTR | <p>La galleta tiene proteínas y azúcar</p> <p>↓</p> <p>tiene proteínas y azúcar</p> <p>La galleta</p> | <p>Devuelve una subcadena desde la aparición de la cadena, excluida o incluida</p> <pre>echo strstr(\$texto,"tiene",false);</pre> <pre>echo strstr(\$texto,"tiene",true);</pre> |
| <ul style="list-style-type: none">• FUNCIÓN CHR | <p>D es la salida</p> | <p>Devuelve el caracter según el código ASCII</p> <pre>echo chr(68)." es la salida";</pre> |



```
if (preg_match("/[aeiou]/i",chr($i))){
echo "\".chr($i).\"N. veces en texto: $val
"; $vocales = $vocales + $val; }
```

El abecedario completo es algo largo y detallarlo exhaustivamente es costoso

"E".N. veces en texto: 1
"a".N. veces en texto: 8
"e".N. veces en texto: 9
"i".N. veces en texto: 2
"o".N. veces en texto: 9
"u".N. veces en texto: 1

Vocales: 30

```
$res = str_replace("o","*", $texto)
```

El abecedario completo es algo largo y detallarlo exhaustivamente es costoso.

El abecedari* c*mplet* es alg* larg* y detallar* exhaustivament es c*st*s*

Todos los derechos reservados

Guillermo Martinez

Métodos

| FUNCIÓN | DESCRIPCIÓN |
|--|---|
| <code>strlen(\$cadena)</code> | Devuelve el número de caracteres de una cadena. |
| <code>substr(\$cadena, \$inicio[, \$largo])</code> | Devuelve una subcadena de una cadena, desde la posición \$inicio y con la longitud \$largo si está presente (si no, hasta el final) |
| <code>strpos(\$cadena, \$subcadena[, \$inicio])</code> | Devuelve la posición de \$subcadena dentro de \$cadena (a partir de \$inicio, si está presente). FALSE ó 0 si no se encuentra. |
| <code>strtoupper(\$cadena)</code> | Devuelve la cadena en mayúsculas. |
| <code>strtolower(\$cadena)</code> | Devuelve la cadena en minúsculas |
| <code>trim(\$cadena)</code> | Devuelve la misma cadena pero eliminando los espacios del principio y el final |
| <code>ltrim(\$cadena[, \$caracteres])</code> | Elimina los espacios del principio del String. Se le puede pasar una lista de caracteres y también los eliminará. |

| | |
|--|--|
| str_replace(\$txtBuscar, \$txtReemplazar, \$cadena[, \$numero]) | Cambia las ocurrencias de \$txtBuscar en \$cadena por \$txtReemplazar (si se indica \$numero, guarda el número de reemplazos realizados) |
| strrev(\$cadena) | Devuelve la cadena invertida. |

1. `addslashes` — Escapa una cadena al estilo de C
2. `addslashes` — Escapa un string con barras invertidas
3. `bin2hex` — Convierte datos binarios en su representación hexadecimal
4. `chop` — Alias de `rtrim`
5. `chr` — Devuelve un caracter específico
6. `chunk_split` — Divide una cadena en trozos más pequeños
7. `convert_cyr_string` — Convierte de un juego de caracteres cirílico a otro juego de caracteres cirílico
8. `convert_uudecode` — Descodifica una cadena codificada mediante `uuencode`
9. `convert_uuencode` — Codificar mediante `uuencode` una cadena
10. `count_chars` — Devuelve información sobre los caracteres usados en una cadena
11. `crc32` — Calcula el polinomio `crc32` de una cadena
12. `crypt` — Hash de cadenas de un sólo sentido
13. `echo` — Muestra una o más cadenas
14. `explode` — Divide un string en varios string
15. `fprintf` — Escribir una cadena con formato a una secuencia
16. `get_html_translation_table` — Devuelve la tabla de traducción utilizada por `htmlspecialchars` y `htmlentities`
17. `hebreu` — Convierte texto hebreo lógico a texto visual
18. `hebreuvc` — Convertir texto de hebreo lógico a texto visual con conversión de línea nueva
19. `hex2bin` — Decodifica una cadena binaria codificada hexadecimalmente
20. `html_entity_decode` — Convierte todas las entidades HTML a sus caracteres correspondientes
21. `htmlentities` — Convierte todos los caracteres aplicables a entidades HTML
22. `htmlspecialchars_decode` — Convierte entidades HTML especiales de nuevo en caracteres
23. `htmlspecialchars` — Convierte caracteres especiales en entidades HTML
24. `implode` — Une elementos de un array en un string
25. `join` — Alias de `implode`
26. `lcfirst` — Pasa a minúscula el primer caracter de un string
27. `levenshtein` — Cálculo de la distancia Levenshtein entre dos strings
28. `localeconv` — Obtener información sobre el formato numérico
29. `ltrim` — Retira espacios en blanco (u otros caracteres) del inicio de un string
30. `md5_file` — Calcula el resumen criptográfico `md5` de un archivo dado
31. `md5` — Calcula el 'hash' `md5` de un string
32. `metaphone` — Calcula la clave metaphone de un string
33. `money_format` — Da formato a un número como un string de moneda
34. `nl_langinfo` — Consulta información sobre el idioma y la configuración regional
35. `nl2br` — Inserta saltos de línea HTML antes de todas las nuevas líneas de un string
36. `number_format` — Formatear un número con los millares agrupados
37. `ord` — Convierte el primer byte de un string a un valor entre 0 y 255
38. `parse_str` — Convierte el string en variables
39. `print` — Mostrar una cadena
40. `printf` — Imprimir una cadena con formato
41. `quoted_printable_decode` — Convierte un string `quoted-printable` en un string de 8 bits
42. `quoted_printable_encode` — Convierte un string de 8 bits en un string `quoted-printable`

43. `quotemeta` — Escapa meta caracteres
44. `rtrim` — Retira los espacios en blanco (u otros caracteres) del final de un string
45. `setlocale` — Establecer la información del localismo
46. `sha1_file` — Calcula el hash sha1 de un archivo
47. `sha1` — Calcula el 'hash' sha1 de un string
48. `similar_text` — Calcula la similitud entre dos strings
49. `soundex` — Calcula la clave soundex de un string
50. `sprintf` — Devuelve un string formateado
51. `sscanf` — Interpreta un string de entrada de acuerdo con un formato
52. `str_contains` — Determine if a string contains a given substring
53. `str_ends_with` — Checks if a string ends with a given substring
54. `str_getcsv` — Convierte un string con formato CSV a un array
55. `str_ireplace` — Versión insensible a mayúsculas y minúsculas de `str_replace`
56. `str_pad` — Rellena un string hasta una longitud determinada con otro string
57. `str_repeat` — Repite un string
58. `str_replace` — Reemplaza todas las apariciones del string buscado con el string de reemplazo
59. `str_rot13` — Realizar la transformación rot13 sobre una cadena
60. `str_shuffle` — Reordena aleatoriamente una cadena
61. `str_split` — Convierte un string en un array
62. `str_starts_with` — Checks if a string starts with a given substring
63. `str_word_count` — Devuelve información sobre las palabras utilizadas en un string
64. `strcasecmp` — Comparación de string segura a nivel binario e insensible a mayúsculas y minúsculas
65. `strchr` — Alias de `strstr`
66. `strcmp` — Comparación de string segura a nivel binario
67. `strcoll` — Comparación de cadenas basada en la localidad
68. `strcspn` — Averiguar la longitud del segmento inicial que no coincida con una máscara
69. `strip_tags` — Retira las etiquetas HTML y PHP de un string
70. `stripclashes` — Desmarca la cadena marcada con `addclashes`
71. `stripos` — Encuentra la posición de la primera aparición de un substring en un string sin considerar mayúsculas ni minúsculas
72. `stripslashes` — Quita las barras de un string con comillas escapadas
73. `stristr` — `strstr` insensible a mayúsculas y minúsculas
74. `strlen` — Obtiene la longitud de un string
75. `strnatcasecmp` — Comparación de strings, insensible a mayúsculas y minúsculas, utilizando un algoritmo de "orden natural"
76. `strnatcmp` — Comparación de strings utilizando un algoritmo de "orden natural"
77. `strncasecmp` — Comparación de los primeros n caracteres de cadenas, segura con material binario e insensible a mayúsculas y minúsculas
78. `strncmp` — Comparación segura a nivel binario de los primeros n caracteres entre strings
79. `strpbrk` — Buscar una cadena por cualquiera de los elementos de un conjunto de caracteres
80. `strpos` — Encuentra la posición de la primera ocurrencia de un substring en un string
81. `strrchr` — Encuentra la última aparición de un caracter en un string
82. `strrev` — Invierte una string

83. `strrpos` — Encuentra la posición de la última aparición de un substring insensible a mayúsculas y minúsculas en un string
84. `strrpos` — Encuentra la posición de la última aparición de un substring en un string
85. `strspn` — Averigua la longitud del segmento inicial de un string que consista únicamente en caracteres contenidos dentro de una máscara dada
86. `strstr` — Encuentra la primera aparición de un string
87. `strtok` — Tokeniza string
88. `strtolower` — Convierte un string a minúsculas
89. `strtoupper` — Convierte un string a mayúsculas
90. `strtr` — Convierte caracteres o reemplaza substrings
91. `substr_compare` — Comparación segura a nivel binario de dos o más strings desde un índice hasta una longitud de caracteres dada
92. `substr_count` — Cuenta el número de apariciones del substring
93. `substr_replace` — Reemplaza el texto dentro de una porción de un string
94. `substr` — Devuelve parte de una cadena
95. `trim` — Elimina espacio en blanco (u otro tipo de caracteres) del inicio y el final de la cadena
96. `ucfirst` — Convierte el primer carácter de una cadena a mayúsculas
97. `ucwords` — Convierte a mayúsculas el primer carácter de cada palabra de una cadena
98. `utf8_decode` — Convierte un string desde UTF-8 a ISO-8859-1, sustituyendo los caracteres no válidos o no representables
99. `utf8_encode` — Convierte una cadena de ISO-8859-1 a UTF-8
100. `vfprintf` — Escribe un string con formato en un flujo
101. `vprintf` — Muestra una cadena con formato
102. `vsprintf` — Devuelve una cadena con formato
103. `wordwrap` — Ajusta un string hasta un número dado de caracteres