

Desarrollo Web en Entorno Servidor

3.- PHP Avanzado.
Funciones predefinidas.

IES Severo Ochoa



Índice

- Funciones predefinidas
- Cadenas
- Matemáticas
- Fechas
- Otras

Funciones predefinidas

- El lenguaje ofrece un abanico de funciones ya definidas.
- <https://www.php.net/manual/es/funcref.php>

Cadenas

- Se puede crear con comillas simples (' ', sin interpretación) o comillas dobles (" " interpretan el contenido)
 - *magic quotes* (" ") interpretan las variables y las secuencias de escape (\n, \t, \\$, {, ...)
 - "Me llamo \$nombre"
 - "Son 30 {\$moneda}s"
- Se acceden a los caracteres como si fuera un array.
- `$cadena = "Yo soy Batman";`
- `$ygriega = $cadena[0];`

Mostrando cadenas

- `echo, print()`
- **`printf`**(\$cadena, \$valor1, \$valor2...)
 - Similar a C, salida formateada
 - https://www.w3schools.com/php/func_string_printf.asp
 - `%d, %c, %.2f, %x, %03d`
- **`sprintf`**
 - Devuelve una cadena con el resultado

Operaciones básicas

- <https://www.php.net/manual/es/ref.strings.php>
- **strlen**: obtiene la longitud de una cadena y devuelve un número entero
- **substr**: devuelve una subcadena de la cadena original
- **str_replace**: reemplaza caracteres en una cadena
- **strtolower** y **strtoupper**
 - Transforman una cadena de caracteres en la misma cadena en minúsculas o mayúsculas respectivamente

Ejemplo cadenas I

```
<?php
$cadena = "El caballero oscuro";
$tam = strlen($cadena);
echo "La longitud de '$cadena' es: $tam <br />";

$oscuro = substr($cadena, 13); // desde 13 al final
$caba = substr($cadena, 3, 4); // desde 3, 4 letras
$katman = str_replace("c", "k", $cadena);
echo "$oscuro $caba ahora es $katman";

echo "Grande ".strtoupper($cadena);
?>
```

Trabajando con caracteres

- **chr**: obtiene el carácter a partir de un ASCII
- **ord**: obtiene el ASCII de un carácter
- **count_chars**: obtiene las apariciones de un carácter en una cadena

```
<?php
function despues(string $letra): string {
    $asciiLetra = ord($letra);
    return chr($asciiLetra + 1);
}
```

```
echo despues("B");
?>
```


Limpiando cadenas

- `trim`: elimina los espacios al principio y al final
 - `ltrim` / `rtrim` o `chop`: Elimina los espacios iniciales / finales de una cadena.
- `str_pad`: rellena la cadenas hasta una longitud especificada y con el carácter o caracteres especificados.

```
<?php
```

```
$cadena = " Programando en PHP ";
```

```
$limpia = trim($cadena); // "Programando en PHP"
```

```
$sucia = str_pad($limpia, 23, "."); // "Programando en PHP....."
```

```
?>
```

Comparando cadenas

- Comparación con `==` o estricta con `===`
- También funcionan `<` y `>` si ambas son cadenas
- Al comparar cadenas con valores numericos:
 - `strcmp`: 0 iguales, `<0` si `a<b` o `>0` si `a>b`
 - `strcasecmp`: las pasa a minúsculas y compara
 - `strncmp` / `strncasecmp`: compara los N primeros caracteres
 - `strnatcmp`: comparaciones naturales

Ejemplo comparando cadenas

```
<?php
```

```
$frase1 = "Alfa";  
$frase2 = "Alfa";  
$frase3 = "Beta";  
$frase4 = "Alfa5";  
$frase5 = "Alfa10";
```

```
var_dump( $frase1 == $frase2 ); // true  
var_dump( $frase1 === $frase2 ); // true  
var_dump( strcmp($frase1, $frase2) ); // 0  
var_dump( strncmp($frase1, $frase5, 3) ); // 0  
var_dump( $frase2 < $frase3 ); // true  
var_dump( strcmp($frase2, $frase3) ); // -1  
var_dump( $frase4 < $frase5 ); // false  
var_dump( strcmp($frase4, $frase5) ); // 4 → f4 > f5  
var_dump( strnatcmp($frase4, $frase5) ); // -1 → f4 < f5  
?>
```

Buscando en cadenas

- `strpos` / `strrpos`: busca en una cadena y devuelve la posición de la primera/última ocurrencia
- `strstr` / `strchr` (alias): busca una cadena y devuelve la cadena a partir de donde la ha encontrado
 - `stristr`: ignora las mayúsculas

```
<?php
```

```
$frase = "Quien busca encuentra, eso dicen, a veces";  
$pos1 = strpos($frase, ","); // encuentra la primera coma  
$pos2 = strrpos($frase, ","); // encuentra la última coma  
$trasComa = strstr($frase, ","); // ", eso dicen, a veces"
```

```
?>
```

Rompiendo cadenas

- **explode**: convierte en array la cadena mediante un separador.
- **implode** / **join**: pasa un array a cadena con un separador
- **str_split** / **chunk_split**: pasa una cadena a una array/cadena cada X caracteres
- Trabajando con tokens
 - 1) **strtok**(cadena, separador)
 - 2) Dentro del bucle: **strtok**(separador)
- **sscanf**: al revés que **printf**, crea un array a partir de la cadena y el patrón.

Ejemplo rompiendo cadenas

```
<?php
$frase = "Quien busca encuentra, eso dicen, a veces";
$partes = explode(",", $frase);

$ciudades = ["Elche", "Aspe", "Alicante"];
$cadenaCiudades = implode(">", $ciudades);

$partes3cadena = chunk_split($frase, 3);
// Qui
// en
// bus
// ca
// ...
$partes3array = str_split($frase, 3);
// ["Qui", "en ", "bus", "ca ", "enc", ...]
?>
```

Operaciones con subcadenas

- `substr_count`: número de veces que aparece la subcadena dentro de la cadena
- `substr_replace`: reemplaza parte de la cadena

```
<?php
$batman = "Bruce Wayne es Batman";
$empresa = substr($batman, 6, 5); // Wayne
$bes = substr_count($batman, "B"); // 2
// Bruce Wayne es camarero
$camarero1 = substr_replace($batman, "camarero", 15);
$camarero2 = substr_replace($batman, "camarero", -6);
// Bruno es Batman
$bruno = substr_replace($batman, "Bruno", 0, 11);
?>
```

Codificaciones

- **htmlentities**: convierte a entidades HTML
 - á → á ñ → ñ < → <
 - **htmlspecialchars**: idem pero solo con los caracteres especiales (&, “, ', <, >)
- **striptags**: elimina etiquetas HTML.
- **n12br**: cambia saltos de línea por
.
- **rawurlencode** / **rawurldecode**:
codifica/decodifica una URL (espacios, ...)

Funciones matemáticas

- <https://www.php.net/manual/es/ref.math.php>
- Constantes ya definidas
 - `M_PI`, `M_E`, `M_EULER`, `M_LN2`, `M_LOG2E`
 - `PHP_INT_MAX`, `PHP_FLOAT_MAX`
- Cálculo
 - `pow`, `sqrt`, `log`, `decbin`, `bindec`,
`decoct`, `dechex`, `base_convert`, `max`, `min`
- Trigonométricas
 - `sin`, `cos`, `tan`, `deg2rad`, `rad2deg`
- Aleatorios
 - `rand`, `mt_rand` // más rápida

Redondeo

- **number_format**: pasa a cadena con decimales y separadores de decimales y de miles

```
<?php
$nf = 1234.5678;
echo number_format($nf, 2); // 1,234.57
echo number_format($nf, 2, "M", "#"); // 1#234M57
?>
```

- **abs**: valor absoluto
- **round, ceil, floor**

```
<?php
$num = 7.7;
$siete = floor($num);
$ocho = ceil($num);

$otro = 4.49;
$cuatro = round($otro);
$cuatrocincos =
round($otro, 1);
$cinco =
round($cuatrocincos);
?>
```

Tipos de datos

- floatval, intval, strval
- settype: fuerza la conversión
- gettype: obtiene el tipo
- is_int, is_float, is_string, is_array, is_object

```
<?php
    $uno = 1;
    var_dump(is_int($uno)); // true
    settype($uno, "string");
    var_dump(is_int($uno)); // false
    var_dump(is_string($uno)); // true
    settype($uno, "float");
    var_dump(is_int($uno)); // false
    var_dump(is_float($uno)); // true
    var_dump(is_int(intval($uno))); // true
?>
```

Funciones de tratamiento de fechas

- <https://www.php.net/manual/es/refs.calendar.php>
- Tiempo Unix: 1/1/70 0:0:0
- **time()**: devuelve número de segundos desde el tiempo Unix respecto la fecha actual → hora local del servidor al cargarse la página.
- **mktime**(\$hora, \$min, \$seg, \$mes, \$dia, \$año): devuelve la marca de tiempo Unix para una fecha concreta.

Fechas formateadas

- `$cadena = date($formato [, $tiempoUnix])`
- Formatos en:
 - <https://www.php.net/manual/es/function.date.php>

```
<?php
$ahora = time();
$cadena = date("d/m/Y");
$mes = date("m");
$manyana = mktime(0, 0, 0, $mes, date("d")+1, date("Y"));
$mesAnterior = mktime(0, 0, 0, $mes-1, date("d"), date("Y"));
$anyoSiguiente = mktime(0, 0, 0, $mes, date("d"), date("Y")+1);

echo $ahora."<br />";
echo $cadena."<br />";
echo date("d-m-Y // H:i:s")."<br />";
echo date("d/m/Y", $manyana);
?>
```

Fechas con localización

- **setlocale**(\$categoria,\$localizacion)
 - \$categoria → CL_TIME.
 - \$localizacion = "" → código del país del servidor
 - `setlocale(LC_Time,"");`
- Si el servidor está en el extranjero:
`setlocale(LC_TIME,"esp,sp,spa,es,spanish") ;`
- Y le ponemos la zona horaria:
`date_default_timezone_set("Europe/Madrid") ;`
- http://www.loc.gov/standards/iso639-2/php/code_list.php
- <http://es.php.net/timezones>

Mostrando fechas

- `strftime ($formato, $tiempo_Unix) ;`
- Muestra la fecha en formato local
- Formatos en:
 - <https://www.php.net/manual/es/function.strftime.php>

```
<?php
$ahora = date("d/F/y");
$ahoraEsp = strftime("%d/%B/%y");
setLocale(LC_TIME, "de_DE", "de_DE.UTF-8");
$ahoraDe = strftime("%d/%B/%y");
?>
```

¿Alguna pregunta?