**date**

(PHP 4, PHP 5)

date — Dar formato a la fecha/hora local

**Descripción** [**¶**](http://php.net/manual/es/function.date.php#refsect1-function.date-description)

string **date** ( string $format [, int $timestamp = time() ] )

Devuelve una cadena formateada según el formato dado usando el parámetro de tipo integer timestamp dado o el momento actual si no se da una marca de tiempo. En otras palabras, timestamp es opcional y por defecto es el valor de [time()](http://php.net/manual/es/function.time.php).

**Parámetros** [**¶**](http://php.net/manual/es/function.date.php#refsect1-function.date-parameters)

format

El formato de la fecha de salida tipo [string](http://php.net/manual/es/language.types.string.php). Vea las opciones de formato más abajo. También hay varias [constantes de fecha predefinidas](http://php.net/manual/es/class.datetime.php#datetime.constants.types) que pueden usarse en su lugar, así por ejemplo **DATE\_RSS** contiene la cadena de formato *'D, d M Y H:i:s'*.

| **Los siguientes caracteres están reconocidos en el parámetro de cadena format** | | |
| --- | --- | --- |
| **Carácter de format** | **Descripción** | **Ejemplo de valores devueltos** |
| *Día* | --- | --- |
| *d* | Día del mes, 2 dígitos con ceros iniciales | *01* a *31* |
| *D* | Una representación textual de un día, tres letras | *Mon* hasta *Sun* |
| *j* | Día del mes sin ceros iniciales | *1* a *31* |
| *l* ('L' minúscula) | Una representación textual completa del día de la semana | *Sunday* hasta *Saturday* |
| *N* | Representación numérica ISO-8601 del día de la semana (añadido en PHP 5.1.0) | *1* (para lunes) hasta *7* (para domingo) |
| *S* | Sufijo ordinal inglés para el día del mes, 2 caracteres | *st*, *nd*, *rd* o *th*. Funciona bien con *j* |
| *w* | Representación numérica del día de la semana | *0* (para domingo) hasta *6* (para sábado) |
| *z* | El día del año (comenzando por 0) | *0* hasta *365* |
| *Semana* | --- | --- |
| *W* | Número de la semana del año ISO-8601, las semanas comienzan en lunes (añadido en PHP 4.1.0) | Ejemplo: *42* (la 42ª semana del año) |
| *Mes* | --- | --- |
| *F* | Una representación textual completa de un mes, como January o March | *January* hasta *December* |
| *m* | Representación numérica de una mes, con ceros iniciales | *01* hasta *12* |
| *M* | Una representación textual corta de un mes, tres letras | *Jan* hasta *Dec* |
| *n* | Representación numérica de un mes, sin ceros iniciales | *1* hasta *12* |
| *t* | Número de días del mes dado | *28* hasta *31* |
| *Año* | --- | --- |
| *L* | Si es un año bisiesto | *1* si es bisiesto, *0* si no. |
| *o* | Número de año ISO-8601. Esto tiene el mismo valor que *Y*, excepto que si el número de la semana ISO (*W*) pertenece al año anterior o siguiente, se usa ese año en su lugar. (añadido en PHP 5.1.0) | Ejemplos: *1999* o *2003* |
| *Y* | Una representación numérica completa de un año, 4 dígitos | Ejemplos: *1999* o *2003* |
| *y* | Una representación de dos dígitos de un año | Ejemplos: *99* o *03* |
| *Hora* | --- | --- |
| *a* | Ante meridiem y Post meridiem en minúsculas | *am* o *pm* |
| *A* | Ante meridiem y Post meridiem en mayúsculas | *AM* o *PM* |
| *B* | Hora Internet | *000* hasta *999* |
| *g* | Formato de 12 horas de una hora sin ceros iniciales | *1* hasta *12* |
| *G* | Formato de 24 horas de una hora sin ceros iniciales | *0* hasta *23* |
| *h* | Formato de 12 horas de una hora con ceros iniciales | *01* hasta *12* |
| *H* | Formato de 24 horas de una hora con ceros iniciales | *00* hasta *23* |
| *i* | Minutos, con ceros iniciales | *00* hasta *59* |
| *s* | Segundos, con ceros iniciales | *00* hasta *59* |
| *u* | Microsegundos (añadido en PHP 5.2.2). Observe que **date()** siempre generará *000000* ya que toma un parámetro de tipo [integer](http://php.net/manual/es/language.types.integer.php), mientras que [DateTime::format()](http://php.net/manual/es/datetime.format.php) admite microsegundos. | Ejemplo: *654321* |
| *Zona Horaria* | --- | --- |
| *e* | Identificador de zona horaria (añadido en PHP 5.1.0) | Ejemplos: *UTC*, *GMT*, *Atlantic/Azores* |
| *I* (i mayúscula) | Si la fecha está en horario de verano o no | *1* si está en horario de verano, *0* si no. |
| *O* | Diferencia de la hora de Greenwich (GMT) en horas | Ejemplo: *+0200* |
| *P* | Diferencia con la hora de Greenwich (GMT) con dos puntos entre horas y minutos (añadido en PHP 5.1.3) | Ejemplo: *+02:00* |
| *T* | Abreviatura de la zona horaria | Ejemplos: *EST*, *MDT* ... |
| *Z* | Índice de la zona horaria en segundos. El índice para zonas horarias al oeste de UTC siempre es negativo, y para aquellas al este de UTC es siempre positivo. | *-43200* hasta *50400* |
| *Fecha/Hora Completa* | --- | --- |
| *c* | Fecha ISO 8601 (añadido en PHP 5) | 2004-02-12T15:19:21+00:00 |
| *r* | Fecha con formato [» RFC 2822](http://www.faqs.org/rfcs/rfc2822) | Ejemplo: *Thu, 21 Dec 2000 16:01:07 +0200* |
| *U* | Segundos desde la Época Unix (1 de Enero del 1970 00:00:00 GMT) | Vea también [time()](http://php.net/manual/es/function.time.php) |

Los caracteres no reconocidos en la cadena de formato serán impresos tal cual. El formato *Z* siempre devolverá *0* cuando se usa [gmdate()](http://php.net/manual/es/function.gmdate.php).

**Nota**:

Ya que esta función sólo acepta marcas de tiempo de tipo [integer](http://php.net/manual/es/language.types.integer.php) el carácter de formato *u* sólo es útil cuando se usa la función [date\_format()](http://php.net/manual/es/function.date-format.php) con marcas de tiempo basadas en usuario creadas con [date\_create()](http://php.net/manual/es/function.date-create.php).

timestamp

El parámetro opcional timestamp es una marca de tiempo Unix de tipo [integer](http://php.net/manual/es/language.types.integer.php) que por defecto es la hora local si no se proporciona ningún valor a timestamp. En otras palabras, es por defecto el valor de la función [time()](http://php.net/manual/es/function.time.php).

**Valores devueltos** [**¶**](http://php.net/manual/es/function.date.php#refsect1-function.date-returnvalues)

Devuelve una cadena de fecha formateada. Si se usa un valor no numérico para timestamp, se devuelve **FALSE** y se emite un error de nivel **E\_WARNING**.

**Errores/Excepciones** [**¶**](http://php.net/manual/es/function.date.php#refsect1-function.date-errors)

Cada vez que se llame a la función date/time se generará un **E\_NOTICE** si la zona horaria no es válida, y/o un mensaje **E\_STRICT** o **E\_WARNING** si se usa la configuración del sistema o la variable global *TZ*. Vea también [date\_default\_timezone\_set()](http://php.net/manual/es/function.date-default-timezone-set.php)

**Historial de cambios** [**¶**](http://php.net/manual/es/function.date.php#refsect1-function.date-changelog)

| **Versión** | **Descripción** |
| --- | --- |
| 5.1.0 | El rango válido de una marca de tiempo es típicamente desde Fri, 13 Dec 1901 20:45:54 GMT a Tue, 19 Jan 2038 03:14:07 GMT. (Estas son las fechas que corresponden a los valores mínimo y máximo de un entero con signo de 32 bit). Sin embargo, antes de PHP 5.1.0 este rango estaba limitado desde 01-01-1970 a 19-01-2038 en algunos sistemas (p.ej. Windows). |
| 5.1.0 | Ahora muestra **E\_STRICT** y **E\_NOTICE** cuando ocurren errores con zonas horarias. |
| 5.1.1 | Hay [constantes](http://php.net/manual/es/datetime.constants.php) útiles de formatos de fecha/hora estándar que se pueden usar para especificar el parámetro format. |

**Ejemplos** [**¶**](http://php.net/manual/es/function.date.php#refsect1-function.date-examples)

**Ejemplo #1 Ejemplo de date()**

<?php  
// Establecer la zona horaria predeterminada a usar. Disponible desde PHP 5.1  
date\_default\_timezone\_set('UTC');  
  
  
// Imprime algo como: Monday  
echo date("l");  
  
// Imprime algo como: Monday 8th of August 2005 03:12:46 PM  
echo date('l jS \of F Y h:i:s A');  
  
// Imprime: July 1, 2000 is on a Saturday  
echo "July 1, 2000 is on a " . date("l", mktime(0, 0, 0, 7, 1, 2000));  
  
/\* Usar las constantes en el parámetro de formato \*/  
// Imprime algo como: Wed, 25 Sep 2013 15:28:57 -0700  
echo date(DATE\_RFC2822);  
  
// Imprime algo como: 2000-07-01T00:00:00+00:00  
echo date(DATE\_ATOM, mktime(0, 0, 0, 7, 1, 2000));  
?>

Puede prevenir que un carácter reconocido en la cadena de formato sea expandido escapándolo con una barra invertida precedente. Si el carácter con una barra invertida es ya una secuencia especial, necesitará escapar también la barra invertida.

**Ejemplo #2 Escapar caracteres en date()**

<?php  
// imprime algo como: Wednesday the 15th  
echo date('l \t\h\e jS');  
?>

Es posible usar **date()** y [mktime()](http://php.net/manual/es/function.mktime.php) juntos para buscar fechas en el futuro o en el pasado.

**Ejemplo #3 Ejemplo de date() y** [**mktime()**](http://php.net/manual/es/function.mktime.php)

<?php  
$mañana        = mktime(0, 0, 0, date("m")  , date("d")+1, date("Y"));  
$mes\_anterior  = mktime(0, 0, 0, date("m")-1, date("d"),   date("Y"));  
$año\_siguiente = mktime(0, 0, 0, date("m"),   date("d"),   date("Y")+1);  
?>

**Nota**:

Esto puede ser más fiable que añadir o sustraer simplemente el número de segundos de un día o mes a una marca de tiempo debido al horario de verano.

Algunos ejemplos de formatear **date()**. Observe que debería escapar cualesquiera otros caracteres, ya que cualquiera que tenga actualmente un significado especial producirá resultados no deseados, y a otros caracteres se les pueden asignar significado en futuras versiones de PHP. Cuando se escapa un carácter, asegúrese de usar comillas simples para prevenir que caracteres como \n se conviertan en nuevas líneas.

**Ejemplo #4 date() Formatting**

<?php  
// Se asume que hoy es March 10th, 2001, 5:16:18 pm, y que estamos en la  
// zona horaria Mountain Standard Time (MST)  
  
$hoy = date("F j, Y, g:i a");                 // March 10, 2001, 5:16 pm  
$hoy = date("m.d.y");                         // 03.10.01  
$hoy = date("j, n, Y");                       // 10, 3, 2001  
$hoy = date("Ymd");                           // 20010310  
$hoy = date('h-i-s, j-m-y, it is w Day');     // 05-16-18, 10-03-01, 1631 1618 6 Satpm01  
$hoy = date('\i\t \i\s \t\h\e jS \d\a\y.');   // it is the 10th day.  
$hoy = date("D M j G:i:s T Y");               // Sat Mar 10 17:16:18 MST 2001  
$hoy = date('H:m:s \m \i\s\ \m\o\n\t\h');     // 17:03:18 m is month  
$hoy = date("H:i:s");                         // 17:16:18  
$hoy = date("Y-m-d H:i:s");                   // 2001-03-10 17:16:18 (el formato DATETIME de MySQL)  
?>

Para formatear fechas en otros lenguajes debería usar las funciones [setlocale()](http://php.net/manual/es/function.setlocale.php) y [strftime()](http://php.net/manual/es/function.strftime.php) en vez de **date()**.

**Notas** [**¶**](http://php.net/manual/es/function.date.php#refsect1-function.date-notes)

**Nota**:

Para generar una marca de tiempo desde una cadena que representa la fecha, puede usar [strtotime()](http://php.net/manual/es/function.strtotime.php). Adicionalmente, algunas bases de datos tienen funciones para convertir formatos de fecha en marcas de tiempo (como la función [» UNIX\_TIMESTAMP](http://dev.mysql.com/doc/mysql/en/date-and-time-functions.html) de MySQL).

**Sugerencia**

La marca de tiempo del inicio de una petición está disponible en [*$\_SERVER['REQUEST\_TIME']*](http://php.net/manual/es/reserved.variables.server.php) desde PHP 5.1.

**Ver también** [**¶**](http://php.net/manual/es/function.date.php#refsect1-function.date-seealso)

* [gmdate()](http://php.net/manual/es/function.gmdate.php) - Formatea una fecha/hora GMT/UTC
* [idate()](http://php.net/manual/es/function.idate.php) - Formatea una fecha/hora local como un entero
* [getdate()](http://php.net/manual/es/function.getdate.php) - Obtener información de la fecha/hora
* [getlastmod()](http://php.net/manual/es/function.getlastmod.php) - Obtiene la hora de la última modificación de la página
* [mktime()](http://php.net/manual/es/function.mktime.php) - Obtener la marca de tiempo Unix de una fecha
* [strftime()](http://php.net/manual/es/function.strftime.php) - Formatea una fecha/hora local según la configuración regional
* [time()](http://php.net/manual/es/function.time.php) - Devuelve la fecha Unix actual
* [strtotime()](http://php.net/manual/es/function.strtotime.php) - Convierte una descripción de fecha/hora textual en Inglés a una fecha Unix
* [Constantes Predefinidas de DateTime](http://php.net/manual/es/class.datetime.php#datetime.constants.types)

[[add a note](http://php.net/manual/add-note.php?sect=function.date&redirect=http://php.net/manual/es/function.date.php)add a note](http://php.net/manual/add-note.php?sect=function.date&redirect=http://php.net/manual/es/function.date.php)

**User Contributed Notes 43 notes**

[up](http://php.net/manual/vote-note.php?id=106974&page=function.date&vote=up)

[down](http://php.net/manual/vote-note.php?id=106974&page=function.date&vote=down)

51

[***Jimmy*** ¶](http://php.net/manual/es/function.date.php#106974)

**2 years ago**

Things to be aware of when using week numbers with years.  
  
<?php  
echo date("YW", strtotime("2011-01-07")); // gives 201101  
echo date("YW", strtotime("2011-12-31")); // gives 201152  
echo date("YW", strtotime("2011-01-01")); // gives 201152 too  
?>  
  
BUT  
  
<?php  
echo date("oW", strtotime("2011-01-07")); // gives 201101  
echo date("oW", strtotime("2011-12-31")); // gives 201152  
echo date("oW", strtotime("2011-01-01")); // gives 201052 (Year is different than previous example)  
?>  
  
Reason:  
Y is year from the date  
o is ISO-8601 year number  
W is ISO-8601 week number of year  
  
Conclusion:  
if using 'W' for the week number use 'o' for the year.

[up](http://php.net/manual/vote-note.php?id=111514&page=function.date&vote=up)

[down](http://php.net/manual/vote-note.php?id=111514&page=function.date&vote=down)

11

[***FiraSEO*** ¶](http://php.net/manual/es/function.date.php#111514)

**1 year ago**

this how you make an HTML5 <time> tag correctly  
  
<?php  
  
echo '<time datetime="'.date('c').'">'.date('Y - m - d').'</time>';  
  
?>  
  
in the "datetime" attribute you should put a machine-readable value which represent time , the best value is a full time/date with ISO 8601 ( date('c') ) ,,, the attr will be hidden from users  
  
and it doesn't really matter what you put as a shown value to the user,, any date/time format is okay !  
  
This is very good for SEO especially search engines like Google .