

PHP - Date & Time

The built-in library of PHP has a wide range of functions that helps in programmatically handling and manipulating date and time information. Date and Time objects in PHP can be created by passing in a string presentation of date/time information, or from the current system's time.

PHP provides the DateTime class that defines a number of methods. In this chapter, we will have a detailed view of the various Date and Time related methods available in PHP.

The date/time features in PHP implements the ISO 8601 calendar, which implements the current leap-day rules from before the Gregorian calendar was in place. The date and time information is internally stored as a 64-bit number.

Getting the Time Stamp with time()

PHP's time() function gives you all the information that you need about the current date and time. It requires no arguments but returns an integer.

```
time(): int
```

The integer returned by time() represents the number of seconds elapsed since midnight GMT on January 1, 1970. This moment is known as the UNIX epoch, and the number of seconds that have elapsed since then is referred to as a time stamp.

```
</>
```

[Open Compiler](#)

```
<?php
    print time();
?>
```

It will produce the following **output** –

```
1699421347
```

We can convert a time stamp into a form that humans are comfortable with.

Converting a Time Stamp with getdate()

The function `getdate()` optionally accepts a time stamp and returns an associative array containing information about the date. If you omit the time stamp, it works with the current time stamp as returned by `time()`.

The following table lists the elements contained in the array returned by `getdate()`.

Sr.No	Key & Description	Example
1	seconds Seconds past the minutes (0-59)	20
2	minutes Minutes past the hour (0 - 59)	29
3	hours Hours of the day (0 - 23)	22
4	mday Day of the month (1 - 31)	11
5	wday Day of the week (0 - 6)	4
6	mon Month of the year (1 - 12)	7
7	year Year (4 digits)	1997
8	yday Day of year (0 - 365)	19
9	weekday Day of the week	Thursday
10	month Month of the year	January
11	0 Timestamp	948370048

Now you have complete control over date and time. You can format this date and time in whatever format you want.

Example

Take a look at this following **example** –



Open Compiler

```
<?php
    $date_array = getdate();

    foreach ( $date_array as $key => $val ){
        print "$key = $val\n";
    }

    $formatted_date = "Today's date: ";
    $formatted_date .= $date_array['mday'] . "-";
    $formatted_date .= $date_array['mon'] . "-";
    $formatted_date .= $date_array['year'];

    print $formatted_date;
?>
```

It will produce the following **output** –

```
seconds = 0
minutes = 38
hours = 6
mday = 8
wday = 3
mon = 11
year = 2023
yday = 311
weekday = Wednesday
month = November
0 = 1699421880
Today's date: 8-11-2023
```

Explore our [latest online courses](#) and learn new skills at your own pace. Enroll and become a certified expert to boost your career.

Converting a Time Stamp with date()

The `date()` function returns a formatted string representing a date. You can exercise an enormous amount of control over the format that `date()` returns with a string argument that you must pass to it.

```
date(string $format, ?int $timestamp = null): string
```

The date() optionally accepts a time stamp if omitted then current date and time will be used. Any other data you include in the format string passed to date() will be included in the return value.

The following table lists the codes that a format string can contain –

Sr.No	Format & Description	Example
1	a 'am' or 'pm' lowercase	pm
2	A 'AM' or 'PM' uppercase	PM
3	d Day of month, a number with leading zeroes	20
4	D Day of week (three letters)	Thu
5	F Month name	January
6	h Hour (12-hour format - leading zeroes)	12
7	H Hour (24-hour format - leading zeroes)	22
8	g Hour (12-hour format - no leading zeroes)	12
9	G Hour (24-hour format - no leading zeroes)	22
10	i Minutes (0 - 59)	23
11	j Day of the month (no leading zeroes)	20
12	l (Lower 'L') Day of the week	Thursday
13	L Leap year ('1' for yes, '0' for no)	1

14	m Month of year (number - leading zeroes)	1
15	M Month of year (three letters)	Jan
16	r The RFC 2822 formatted date	Thu, 21 Dec 2000 16:01:07 +0200
17	n Month of year (number - no leading zeroes)	2
18	s Seconds of hour	20
19	U Time stamp	948372444
20	y Year (two digits)	06
21	Y Year (four digits)	2006
22	z Day of year (0 - 365)	206
23	Z Offset in seconds from GMT	+5

Example

Take a look at this following **example** –

</>

Open Compiler

```
<?php
    print date("m/d/y G.i:s \n", time()) . PHP_EOL;
    print "Today is ";
    print date("j of F Y, \a\\t g.i a", time());
?>
```

It will produce the following **output** –

11/08/23 11.23:08

Today is 8 2023f November 2023, at 11.23 am

Hope you have good understanding on how to format date and time according to your requirement. For your reference a complete list of all the date and time functions is given in [PHP Date & Time Functions](#).