



# JavaScript Date Formats

◀ Previous

Next ▶

## JavaScript Date Input

There are generally 3 types of JavaScript date input formats:

Type	Example
ISO Date	"2015-03-25" (The International Standard)
Short Date	"03/25/2015"
Long Date	"Mar 25 2015" or "25 Mar 2015"

The ISO format follows a strict standard in JavaScript.

The other formats are not so well defined and might be browser specific.

## JavaScript Date Output

Independent of input format, JavaScript will (by default) output dates in full text string format:

```
Tue Mar 24 2015 19:30:00 GMT-0430 (VET)
```

## JavaScript ISO Dates

ISO 8601 is the international standard for the representation of dates and times.

The ISO 8601 syntax (YYYY-MM-DD) is also the preferred JavaScript date format:

Example (Complete date)

```
var d = new Date("2015-03-25");
```

Try it Yourself »

The computed date will be relative to your time zone.  
Depending on your time zone, the result above will vary between March 24 and March 25.

## ISO Dates (Year and Month)

ISO dates can be written without specifying the day (YYYY-MM):

### Example

```
var d = new Date("2015-03");
```

Try it Yourself »

Time zones will vary the result above between February 28 and March 01.

## ISO Dates (Only Year)

ISO dates can be written without month and day (YYYY):

### Example

```
var d = new Date("2015");
```

Try it Yourself »

Time zones will vary the result above between December 31 2014 and January 01 2015.

## ISO Dates (Date-Time)

ISO dates can be written with added hours, minutes, and seconds (YYYY-MM-DDTHH:MM:SSZ):

### Example

```
var d = new Date("2015-03-25T12:00:00Z");
```

Try it Yourself »

Date and time is separated with a capital T.

UTC time is defined with a capital letter Z.

If you want to modify the time relative to UTC, remove the Z and add +HH:MM or -HH:MM instead:

### Example

```
var d = new Date("2015-03-25T12:00:00-06:30");
```

Try it Yourself »

UTC (Universal Time Coordinated) is the same as GMT (Greenwich Mean Time).

Omitting T or Z in a date-time string can give different result in different browser.

## Time Zones

When setting a date, without specifying the time zone, JavaScript will use the browser's time zone.

When getting a date, without specifying the time zone, the result is converted to the browser's time zone.

In other words: If a date/time is created in GMT (Greenwich Mean Time), the date/time will be converted to CDT (Central US Daylight Time) if a user browses from central US.

## JavaScript Short Dates.

Short dates are written with an "MM/DD/YYYY" syntax like this:

### Example

```
var d = new Date("03/25/2015");
```

Try it Yourself »

## WARNINGS !

In some browsers, months or days with no leading zeroes may produce an error:

```
var d = new Date("2015-3-25");
```

The behavior of "YYYY/MM/DD" is undefined.  
Some browsers will try to guess the format. Some will return NaN.

```
var d = new Date("2015/03/25");
```

The behavior of "DD-MM-YYYY" is also undefined.  
Some browsers will try to guess the format. Some will return NaN.

```
var d = new Date("25-03-2015");
```

## JavaScript Long Dates.

Long dates are most often written with a "MMM DD YYYY" syntax like this:

### Example

```
var d = new Date("Mar 25 2015");
```

Try it Yourself »

Month and day can be in any order:

### Example

```
var d = new Date("25 Mar 2015");
```

Try it Yourself »

And, month can be written in full (January), or abbreviated (Jan):

### Example

```
var d = new Date("January 25 2015");
```

Try it Yourself »

### Example

```
var d = new Date("Jan 25 2015");
```

Try it Yourself »

Commas are ignored. Names are case insensitive:

### Example

```
var d = new Date("JANUARY, 25, 2015");
```

Try it Yourself »

## Date Input - Parsing Dates

If you have a valid date string, you can use the `Date.parse()` method to convert it to milliseconds.

`Date.parse()` returns the number of milliseconds between the date and January 1, 1970:

### Example

```
var msec = Date.parse("March 21, 2012");  
document.getElementById("demo").innerHTML = msec;
```

Try it Yourself »

You can then use the number of milliseconds to **convert it to a date** object:

### Example

```
var msec = Date.parse("March 21, 2012");  
var d = new Date(msec);  
document.getElementById("demo").innerHTML = d;
```

Try it Yourself »

◀ Previous

Next ▶