

## El objeto Date

Los objetos de tipo Date nos sirven para manejar fechas, teniendo en cuenta que en JavaScript las fechas se expresan como el número de milisegundos transcurridos desde el 1 de enero de 1970 a las 00:00:00 UTC (esta fecha se conoce como origen o época UNIX). El rango de fechas posible es de aproximadamente +/- cien millones de días a partir del origen (unos 273972 años).

Para crear un objeto Date:

```
NombreObjetoDate = new Date([parámetros])
```

Donde NombreObjetoDate es el nombre del objeto Date que se va a crear.

Los parámetros en la sintaxis precedente pueden ser cualesquiera de los siguientes:

- **Nada:** crea la fecha de hoy día y la hora. Por ejemplo, hoydía = new Date().
- **Una cadena representando una fecha en la siguiente forma:** "Mes día, año horas:minutos:segundos." Por ejemplo, Navidad95 = new Date("December 25, 1995 13:30:00"). Si se omite las horas, minutos, o segundos, el valor será configurado como cero.
- **Un conjunto de valores enteros para año, mes y día:** Por ejemplo, Navidad95 = new Date(1995,11,25).
- **Un conjunto de valores enteros para año, mes, día, hora, minuto y segundos:** Por ejemplo, Navidad95 = new Date(1995,11,25,9,30,0).

JavaScript manipula las fechas de forma similar a Java. Los dos lenguajes poseen muchos de los mismos métodos para fechas y ambos lenguajes almacenas las fechas como un número de milisegundos desde Enero 1, 1970, 00:00:00.

Donde todos los parámetros son números enteros. El año debe expresarse con 4 cifras. El mes es un entero entre 0 (enero) y 11 (diciembre). El día de la fecha es un entero entre 1 y 31. El día de la semana es un entero entre 0 (domingo) a 6 (sábado). Horas entre 0 y 23. Minutos y segundos entre 0 y 59.

Si no se aporta ninguno de los argumentos, JavaScript crea un objeto con la fecha del sistema en ese instante.

En la siguiente tabla se recogen algunos de los métodos más interesantes de los objetos de tipo Date.

Método	Descripción
getDate() getUTCDate()	Devuelve el día del mes (1-31).
getDay() getUTCDay()	Devuelve el día de la semana (0-6). El 0 corresponde al domingo y el 6 al sábado. Atención, el día no se puede establecer, sólo leer; en otras palabras, no existe setDay().
getFullYear() getUTCFullYear()	Devuelve el año expresado con 4 dígitos. No usar el método getYear() para evitar problemas derivados del efecto 2000 (Y2K).
getHours() getUTCHours()	Devuelve la hora (0-23).
getMilliseconds() getUTCMilliseconds()	Devuelve los milisegundos (0-999).
getMinutes() getUTCMinutes()	Devuelve los minutos (0-59).
getMonth() getUTCMonth()	Devuelve el mes (0-11).
getSeconds() getUTCSeconds()	Devuelve los segundos (0-59).
getTime()	Devuelve los milisegundos transcurridos desde la época UNIX.
parse(fecha)	Devuelve la cantidad de milisegundos transcurridos desde el 1 de enero de 1970 00:00:00 hasta la fecha pasada en el argumento fecha.

setDate( <i>diaMes</i> ) setUTCDate( <i>diaMes</i> )	Establece el día del mes (1-31).
setFullYear( <i>año</i> ) setUTCFullYear( <i>año</i> )	Establece el año con 4 dígitos.
setHours( <i>hora</i> ) setUTCHours( <i>hora</i> )	Establece la hora (0-23).
setMilliseconds( <i>miliseg</i> ) setUTCMilliseconds( <i>miliseg</i> )	Establece los milisegundos (0-999).
setMinutes( <i>minuto</i> ) setUTCMinutes( <i>minuto</i> )	Establece los minutos (0-59).
setMonth( <i>mes</i> ) setUTCMonth( <i>mes</i> )	Establece el mes (0-11).
setSeconds( <i>segundo</i> ) setUTCSeconds( <i>segundo</i> )	Establece los segundos (0-59).
setTime( <i>milisegundos</i> )	Establece la fecha como el número de milisegundos transcurrido desde la época UNIX.
toDateString()	Convierte la fecha del objeto DATE en una cadena de caracteres.
toLocaleDateString()	Devuelve una cadena de la fecha (día de la semana, día, mes y año) en el formato de idioma local que tenga configurado el cliente.
toLocaleString()	Devuelve una cadena de la fecha completa en el formato de idioma local que tenga configurado el cliente.
toLocaleTimeString()	Devuelve una cadena de la hora (hora, minuto, segundo) en el formato de idioma local que tenga configurado el cliente.
toTimeString()	Convierte la parte de tiempo de un objeto Date en una cadena.
toUTCString()	Devuelve la representación de la fecha como una cadena de caracteres en formato UTC. Por ejemplo: Tue, 19 Jun 2012 14:22:01 UTC

El **tiempo universal coordinado** o **UTC** (de un compromiso entre la versión en *inglés*: ***Coordinated Universal Time*** y la versión en *francés*: ***Temps Universel Coordonné***) es el principal *estándar de tiempo* por el cual el mundo regula los relojes y el tiempo. Es uno de los varios sucesores estrechamente relacionados con el *tiempo medio de Greenwich* (GMT). Para la mayoría de los propósitos comunes, UTC es sinónimo de GMT, pero GMT ya no es el estándar definido con más precisión para la comunidad científica.