Date

创建 Date 实例用来处理日期和时间。Date 对象基于1970年1月1日（世界标准时间）起的毫秒数。

new Date();

new Date(value);

new Date(dateString);

new Date(year, month[, day[, hour[, minutes[, seconds[, milliseconds]]]]]);

*需要注意的是只能通过调用 Date 构造函数来实例化日期对象：以常规函数调用它（即不加 new 操作符）将会返回一个字符串，而不是一个日期对象。另外，不像其他JavaScript 类型，Date 对象没有字面量格式。*

*当Date作为构造函数调用并传入多个参数时，如果数值大于合理范围时（如月份为13或者分钟数为70），相邻的数值会被调整。比如 new Date(2013, 13, 1)等于new Date(2014, 1, 1)，它们都表示日期2014-02-01（注意月份是从0开始的）。其他数值也是类似，new Date(2013, 2, 1, 0, 70)等于new Date(2013, 2, 1, 1, 10)，都表示时间2013-03-01T01:10:00。*

*当Date作为构造函数调用并传入多个参数时，所定义参数代表的是当地时间。如果需要世界协调时，使用 new Date({{jsxref("Date.UTC()", "Date.UTC(...)")}}) 和相同参数*

*Date.parse 和Date.getTime 在返回的毫秒数上有区别，parse把微秒去掉了，只精确到秒，但是getTime或其他方式获取毫秒数会精确到微秒数。*

*Date.prototype.getDate()*

*根据本地时间返回指定日期对象的月份中的第几天（1-31）。*

*Date.prototype.getDay()*

*根据本地时间返回指定日期对象的星期中的第几天（0-6）。*

*Date.prototype.getMonth()*

*根据本地时间返回指定日期对象的月份（0-11）。月份从0开始*

属性

Date.prototype

允许为 Date 实例对象添加属性。

Date.length

Date.length 的值是 7。这是该构造函数可接受的参数个数。

方法

Date.now()

返回自 1970-1-1 00:00:00 UTC (世界标准时间)至今所经过的毫秒数。

Date.parse()

解析一个表示日期的字符串，并返回从 1970-1-1 00:00:00 所经过的毫秒数。

Date.UTC()

接受和构造函数最长形式的参数相同的参数（从2到7），并返回从 1970-01-01 00:00:00 UTC 开始所经过的毫秒数。