JavaScript中的Date对象

# Date对象

<http://www.runoob.com/jsref/jsref-obj-date.html>

## Date对象功能

Date 对象用于处理日期和时间。

## 创建Date对象

可以通过 **new 关键词**来定义 Date 对象，有四种方式初始化Date对象：

### new Date() // 当前日期和时间

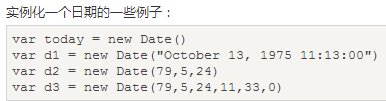
### new Date(milliseconds) //返回从 1970 年 1 月 1 日至今的毫秒数

### new Date(dateString)

### new Date(year, month, day, hours, minutes, seconds, milliseconds)

参数依次为：年月日时分秒毫秒

上面的参数大多数都是可选的，在不指定的情况下，默认参数是0。



## GMT、UTC

GMT（**Greenwich** Mean Time）——格林尼治标准时间

UTC (Universal Time Coordinated)协调世界时，又称世界统一时间、世界标准时间、国际协调时间。

UTC是协调世界时(Universal Time Coordinated)英文缩写，是由国际无线电咨询委员会规定和推荐,并由国际时间局(BIH)负责保持的以秒为基础的时间标度。UTC相当于**本初子午线**(即经度0度)上的**平均太阳时**，过去曾用**格林威治平均时(GMT)**来表示.北京时间比UTC时间早8小时，以1999年1月1日0000UTC为例，UTC时间是零点，北京时间为1999年1月1日早上8点整。

整个地球分为二十四时区，每个时区都有自己的本地时间。在国际无线电通信场合，为了统一起见，使用一个统一的时间，称为通用协调时(UTC, Universal Time Coordinated)。**UTC与格林尼治平均时(GMT, Greenwich Mean Time)一样，都与英国伦敦的本地时相同。**

**格林尼治标准时间**（GMT，旧译“格林威治平均时间”或“格林威治标准时间”）是指位于伦敦郊区的皇家格林尼治天文台的标准时间，因为本初子午线被定义在通过那里的经线。

**理论上来说，格林尼治标准时间的正午是指当太阳横穿格林尼治子午线时（也就是在格林尼治上空最高点时）的时间。**由于地球在它的椭圆轨道里的运动速度不均匀，这个时刻可能和实际的太阳时相差16分钟。

**地球每天的自转是有些不规则的，而且正在缓慢减速。所以，格林尼治时间已经不再被作为标准时间使用。**现在的标准时间——协调世界时（UTC）——由原子钟提供。

自1924年2月5日开始，格林尼治天文台每隔一小时会向全世界发放调时信息。

协调世界时(UTC)

英文：Coordinated Universal Time

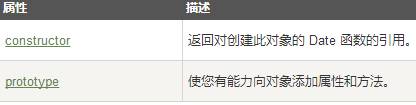
法文：Temps universel coordonné

别称：世界统一时间， 世界标准时间， 国际协调时间

协调世界时，又称世界统一时间，世界标准时间，国际协调时间，简称UTC。它从英文“Coordinated Universal Time”／法文“Temps Universel Cordonné”而来。

协调世界时是以原子时秒长为基础，在时刻上尽量接近于世界时的一种时间计量系统。

## Date对象的属性



## Date对象的方法

### getXxx方法

获取方法getXxx返回指定的数值：

年：getFullYear，四位数字；

月：getMonth()，0-11，注意0表示1月份。

日：getDate()，1-31；

星期：getDay()，0-6，注意0表示星期一。

时：getHours()，0-23；

分：getMinutes()，0-59；

秒：getSeconds()，0-59；

毫秒：getMilliseconds()，0-999；

返回总的毫秒数：getTime(),返回 1970 年 1 月 1 日至今的毫秒数。





### setXxx方法





### toXxx方法



### parse方法：返回毫秒数



### UTC()方法：返回毫秒数



### valueOf()



## 两个Date对象之间的比较

**日期对象**可直接利用比较运算符进行大小比较。

## 获取星期

getDay() 方法可返回一周（0~6）的某一天的数字。

**注意： 星期天为 0, 星期一为 1, 以此类推**。

获取星期，返回的是0-6的数值，因此需要构造一个数组，才能真正获取星期。

var weeks = ['**星期日**','星期一','星期二','星期三','星期四','星期五','星期六'];

alert(weeks[**new Date().getDay()**]);