

JavaScript Date（日期）对象

日期对象用于处理日期和时间。



在线实例

[如何使用 Date\(\) 方法获得当日的日期。](#)

[getFullYear\(\)](#)

使用 getFullYear() 获取年份。

[getTime\(\)](#)

getTime() 返回从 1970 年 1 月 1 日至今的毫秒数。

[setFullYear\(\)](#)

如何使用 setFullYear() 设置具体的日期。

[toUTCString\(\)](#)

如何使用 toUTCString() 将当日的日期（根据 UTC）转换为字符串。

[getDay\(\)](#)

如何使用 getDay() 和数组来显示星期，而不仅仅是数字。

[Display a clock](#)

如何在网页上显示一个钟表。

完整的 Date 对象参考手册

我们提供 JavaScript Date 对象参考手册，其中包括所有可用于日期对象的属性和方法。[JavaScript Date 对象参考手册](#)。
该手册包含了对每个属性和方法的详细描述以及相关实例。

创建日期

Date 对象用于处理日期和时间。

可以通过 new 关键词来定义 Date 对象。以下代码定义了名为 myDate 的 Date 对象：

有四种方式初始化日期：

```
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)
```

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。

实例化一个日期的一些例子：

```
var today = new Date()
var dl = new Date("October 13, 1975 11:13:00")
```

```
var d2 = new Date(79,5,24)
var d3 = new Date(79,5,24,11,33,0)
```

设置日期

通过使用针对日期对象的方法，我们可以很容易地对日期进行操作。

在下面的例子中，我们为日期对象设置了一个特定的日期 (2010 年 1 月 14 日)：

```
var myDate=new Date();
myDate.setFullYear(2010,0,14);
```

在下面的例子中，我们将日期对象设置为 5 天后的日期：

```
var myDate=new Date();
myDate.setDate(myDate.getDate()+5);
```

注意: 如果增加天数会改变月份或者年份，那么日期对象会自动完成这种转换。

两个日期比较

日期对象也可用于比较两个日期。

下面的代码将当前日期与 2100 年 1 月 14 日做了比较：

```
var x=new Date();
x.setFullYear(2100,0,14);
var today = new Date();

if (x>today)
{
    alert("今天是2100年1月14日之前");
}
else
{
    alert("今天是2100年1月14日之后");
}
```

← JavaScript 字符串 (String) 对象

JavaScript Array (数组) 对象 →

 点我分享笔记