

# JavaScript 字符串

JavaScript 字符串用于存储和处理文本。

## JavaScript 字符串

字符串可以存储一系列字符，如 "John Doe"。

字符串可以是插入到引号中的任何字符。你可以使用单引号或双引号：

### 实例

```
var carname = "Volvo XC60";  
var carname = 'Volvo XC60';
```

你可以使用索引位置来访问字符串中的每个字符：

### 实例

```
var character = carname[7];
```

字符串的索引从 0 开始，这意味着第一个字符索引值为 [0],第二个为 [1], 以此类推。

你可以在字符串中使用引号，字符串中的引号不要与字符串的引号相同：

### 实例

```
var answer = "It's alright";  
var answer = "He is called 'Johnny'";  
var answer = 'He is called "Johnny"';
```

你也可以在字符串添加转义字符来使用引号：

### 实例

```
var x = 'It\'s alright';  
var y = "He is called \"Johnny\"";
```

[尝试一下 »](#)

## 字符串长度

可以使用内置属性 **length** 来计算字符串的长度：

### 实例

```
var txt = "ABCDEFGHJKLMNOPQRSTUVWXYZ";  
var sln = txt.length;
```

[尝试一下 »](#)

# 特殊字符

在 JavaScript 中，字符串写在单引号或双引号中。

因为这样，以下实例 JavaScript 无法解析：

```
"We are the so-called "Vikings" from the north."
```

字符串 "We are the so-called " 被截断。

如何解决以上的问题呢？可以使用反斜杠 (\) 来转义 "Vikings" 字符串中的双引号，如下：

```
"We are the so-called \"Vikings\" from the north."
```

反斜杠是一个**转义字符**。转义字符将特殊字符转换为字符串字符：

转义字符 (\) 可以用于转义撇号，换行，引号，等其他特殊字符。

下表中列举了在字符串中可以使用转义字符转义的特殊字符：

代码	输出
\'	单引号
\"	双引号
\\	反斜杠
\n	换行
\r	回车
\t	tab(制表符)
\b	退格符
\f	换页符

# 字符串可以是对象

通常，JavaScript 字符串是原始值，可以使用字符创建：`var firstName = "John"`

但我们也可以使用 new 关键字将字符串定义为一个对象：`var firstName = new String("John")`

## 实例

```
var x = "John";
var y = new String("John");
typeof x // 返回 String
typeof y // 返回 Object
```

尝试一下 »

不要创建 String 对象。它会拖慢执行速度，并可能产生其他副作用：



实例

```
var x = "John";
var y = new String("John");
(x === y) // 结果为 false, 因为 x 是字符串, y 是对象
```

尝试一下 »

=== 为绝对相等，即数据类型与值都必须相等。

字符串属性和方法

原始值字符串，如 "John", 没有属性和方法(因为他们不是对象)。  
原始值可以使用 JavaScript 的属性和方法，因为 JavaScript 在执行方法和属性时可以把原始值当作对象。  
字符串方法我们将在下一章节中介绍。

字符串属性

属性	描述
constructor	返回创建字符串属性的函数
length	返回字符串的长度
prototype	允许您向对象添加属性和方法

字符串方法

更多方法实例可以参见：[JavaScript String 对象](#)。

方法	描述
charAt()	返回指定索引位置的字符
charCodeAt()	返回指定索引位置字符的 Unicode 值
concat()	连接两个或多个字符串，返回连接后的字符串
fromCharCode()	将 Unicode 转换为字符串
indexOf()	返回字符串中检索指定字符第一次出现的位置
lastIndexOf()	返回字符串中检索指定字符最后一次出现的位置

localeCompare()	用本地特定的顺序来比较两个字符串
match()	找到一个或多个正则表达式的匹配
replace()	替换与正则表达式匹配的子串
search()	检索与正则表达式相匹配的值
slice()	提取字符串的片断，并在新的字符串中返回被提取的部分
split()	把字符串分割为子字符串数组
substr()	从起始索引号提取字符串中指定数目的字符
substring()	提取字符串中两个指定的索引号之间的字符
toLocaleLowerCase()	根据主机的语言环境把字符串转换为小写，只有几种语言（如土耳其语）具有地方特有的大小写映射
toLocaleUpperCase()	根据主机的语言环境把字符串转换为大写，只有几种语言（如土耳其语）具有地方特有的大小写映射
toLowerCase()	把字符串转换为小写
toString()	返回字符串对象值
toUpperCase()	把字符串转换为大写
trim()	移除字符串首尾空白
valueOf()	返回某个字符串对象的原始值

+

3 篇笔记

✎ 写笔记