

JavaScript 语法一览

JavaScript 对象

对象	描述
Array	Array 对象用于在单个的变量中存储多个值。
Boolean	Boolean 对象表示两个值: "true" 或 "false"。
Date	Date 对象用于处理日期和时间。
Math	Math 对象用于执行数学任务。
Number	Number 对象是原始数值的包装对象。
String	String 对象用于处理文本（字符串）。
RegExp	RegExp 对象表示正则表达式，它是对字符串执行模式匹配的强大工具。

Array 对象属性

属性	描述
constructor	返回对创建此对象的数组函数的引用。
length	设置或返回数组中元素的数目。
prototype	使您有能力向对象添加属性和方法。

Array 对象方法

方法	描述
concat()	连接两个或更多的数组，并返回结果。
join()	把数组的所有元素放入一个字符串。元素通过指定的分隔符进行分隔。
pop()	删除并返回数组的最后一个元素
push()	向数组的末尾添加一个或更多元素，并返回新的长度。
reverse()	颠倒数组中元素的顺序。
shift()	删除并返回数组的第一个元素
slice()	从某个已有的数组返回选定的元素
sort()	对数组的元素进行排序
splice()	删除元素，并向数组添加新元素。
toSource()	返回该对象的源代码。
toString()	把数组转换为字符串，并返回结果。

<code>toLocaleString()</code>	把数组转换为本地数组，并返回结果。
<code>unshift()</code>	向数组的开头添加一个或更多元素，并返回新的长度。
<code>valueOf()</code>	返回数组对象的原始值

Boolean 对象属性

属性	描述
<code>constructor</code>	返回对创建此对象的 <code>Boolean</code> 函数的引用
<code>prototype</code>	使您有能力向对象添加属性和方法。

Boolean 对象方法

方法	描述
<code>toSource()</code>	返回该对象的源代码。
<code>toString()</code>	把逻辑值转换为字符串，并返回结果。
<code>valueOf()</code>	返回 <code>Boolean</code> 对象的原始值。

Date 对象属性

属性	描述
<code>constructor</code>	返回对创建此对象的 <code>Date</code> 函数的引用。
<code>prototype</code>	使您有能力向对象添加属性和方法。

Date 对象方法

方法	描述
<code>Date()</code>	返回当日的日期和时间。
<code>getDate()</code>	从 <code>Date</code> 对象返回一个月中的某一天（1 ~ 31）。
<code>getDay()</code>	从 <code>Date</code> 对象返回一周中的某一天（0 ~ 6）。
<code>getMonth()</code>	从 <code>Date</code> 对象返回月份（0 ~ 11）。
<code>getFullYear()</code>	从 <code>Date</code> 对象以四位数字返回年份。
<code>getYear()</code>	请使用 <code>getFullYear()</code> 方法代替。
<code>getHours()</code>	返回 <code>Date</code> 对象的小时（0 ~ 23）。
<code>getMinutes()</code>	返回 <code>Date</code> 对象的分钟（0 ~ 59）。
<code>getSeconds()</code>	返回 <code>Date</code> 对象的秒数（0 ~ 59）。
<code>getMilliseconds()</code>	返回 <code>Date</code> 对象的毫秒（0 ~ 999）。
<code>getTime()</code>	返回 1970 年 1 月 1 日至今的毫秒数。
<code>getTimezoneOffset()</code>	返回本地时间与格林威治标准时间（GMT）的分钟差。

<code>getUTCDate()</code>	根据世界时从 <code>Date</code> 对象返回月中的一天 (1 ~ 31)。
<code>getUTCDay()</code>	根据世界时从 <code>Date</code> 对象返回周中的一天 (0 ~ 6)。
<code>getUTCMonth()</code>	根据世界时从 <code>Date</code> 对象返回月份 (0 ~ 11)。
<code>getUTCFullYear()</code>	根据世界时从 <code>Date</code> 对象返回四位数的年份。
<code>getUTCHours()</code>	根据世界时返回 <code>Date</code> 对象的小时 (0 ~ 23)。
<code>getUTCMinutes()</code>	根据世界时返回 <code>Date</code> 对象的分钟 (0 ~ 59)。
<code>getUTCSeconds()</code>	根据世界时返回 <code>Date</code> 对象的秒钟 (0 ~ 59)。
<code>getUTCMilliseconds()</code>	根据世界时返回 <code>Date</code> 对象的毫秒 (0 ~ 999)。
<code>parse()</code>	返回 1970 年 1 月 1 日午夜到指定日期 (字符串) 的毫秒数。
<code>setDate()</code>	设置 <code>Date</code> 对象中月的某一天 (1 ~ 31)。
<code>setMonth()</code>	设置 <code>Date</code> 对象中月份 (0 ~ 11)。
<code>setFullYear()</code>	设置 <code>Date</code> 对象中的年份 (四位数字)。
<code>setYear()</code>	请使用 <code>setFullYear()</code> 方法代替。
<code>setHours()</code>	设置 <code>Date</code> 对象中的小时 (0 ~ 23)。
<code>setMinutes()</code>	设置 <code>Date</code> 对象中的分钟 (0 ~ 59)。
<code>setSeconds()</code>	设置 <code>Date</code> 对象中的秒钟 (0 ~ 59)。
<code>setMilliseconds()</code>	设置 <code>Date</code> 对象中的毫秒 (0 ~ 999)。
<code>setTime()</code>	以毫秒设置 <code>Date</code> 对象。
<code>setUTCDate()</code>	根据世界时设置 <code>Date</code> 对象中月份的一天 (1 ~ 31)。
<code>setUTCMonth()</code>	根据世界时设置 <code>Date</code> 对象中的月份 (0 ~ 11)。
<code>setUTCFullYear()</code>	根据世界时设置 <code>Date</code> 对象中的年份 (四位数字)。
<code>setUTCHours()</code>	根据世界时设置 <code>Date</code> 对象中的小时 (0 ~ 23)。
<code>setUTCMinutes()</code>	根据世界时设置 <code>Date</code> 对象中的分钟 (0 ~ 59)。
<code>setUTCSeconds()</code>	根据世界时设置 <code>Date</code> 对象中的秒钟 (0 ~ 59)。
<code>setUTCMilliseconds()</code>	根据世界时设置 <code>Date</code> 对象中的毫秒 (0 ~ 999)。
<code>toSource()</code>	返回该对象的源代码。
<code>toString()</code>	把 <code>Date</code> 对象转换为字符串。
<code>toTimeString()</code>	把 <code>Date</code> 对象的时间部分转换为字符串。
<code>toDateString()</code>	把 <code>Date</code> 对象的日期部分转换为字符串。
<code>toGMTString()</code>	请使用 <code>toUTCString()</code> 方法代替。
<code>toUTCString()</code>	根据世界时, 把 <code>Date</code> 对象转换为字符串。
<code>toLocaleString()</code>	根据本地时间格式, 把 <code>Date</code> 对象转换为字符串。
<code>toLocaleTimeString()</code>	根据本地时间格式, 把 <code>Date</code> 对象的时间部分转换为字符串。
<code>toLocaleDateString()</code>	根据本地时间格式, 把 <code>Date</code> 对象的日期部分转换为字符串。
<code>UTC()</code>	根据世界时返回 1970 年 1 月 1 日到指定日期的毫秒数。
<code>valueOf()</code>	返回 <code>Date</code> 对象的原始值。

Math 对象属性

属性	描述
E	返回算术常量 e，即自然对数的底数（约等于 2.718）。
LN2	返回 2 的自然对数（约等于 0.693）。
LN10	返回 10 的自然对数（约等于 2.302）。
LOG2E	返回以 2 为底的 e 的对数（约等于 1.414）。
LOG10E	返回以 10 为底的 e 的对数（约等于 0.434）。
PI	返回圆周率（约等于 3.14159）。
SQRT1_2	返回返回 2 的平方根的倒数（约等于 0.707）。
SQRT2	返回 2 的平方根（约等于 1.414）。

Math 对象方法

方法	描述
abs(x)	返回数的绝对值。
acos(x)	返回数的反余弦值。
asin(x)	返回数的反正弦值。
atan(x)	以介于 $-\pi/2$ 与 $\pi/2$ 弧度之间的数值来返回 x 的反正切值。
atan2(y,x)	返回从 x 轴到点 (x,y) 的角度(介于 $-\pi/2$ 与 $\pi/2$ 弧度之间)。
ceil(x)	对数进行上舍入。
cos(x)	返回数的余弦。
exp(x)	返回 e 的指数。
floor(x)	对数进行下舍入。
log(x)	返回数的自然对数（底为 e）。
max(x,y)	返回 x 和 y 中的最高值。
min(x,y)	返回 x 和 y 中的最低值。
pow(x,y)	返回 x 的 y 次幂。
random()	返回 0 ~ 1 之间的随机数。
round(x)	把数四舍五入为最接近的整数。
sin(x)	返回数的正弦。
sqrt(x)	返回数的平方根。
tan(x)	返回角的正切。
toSource()	返回该对象的源代码。
valueOf()	返回 Math 对象的原始值。

Number 对象属性

属性	描述
constructor	返回对创建此对象的 Number 函数的引用。
MAX_VALUE	可表示的最大的数。
MIN_VALUE	可表示的最小的数。
NaN	非数字值。
NEGATIVE_INFINITY	负无穷大，溢出时返回该值。
POSITIVE_INFINITY	正无穷大，溢出时返回该值。
prototype	使您有能力向对象添加属性和方法。

Number 对象方法

方法	描述
toString	把数字转换为字符串，使用指定的基数。
toLocaleString	把数字转换为字符串，使用本地数字格式顺序。
toFixed	把数字转换为字符串，结果的小数点后有指定位数的数字。
toExponential	把对象的值转换为指数计数法。
toPrecision	把数字格式化为指定的长度。
valueOf	返回一个 Number 对象的基本数字值。

String 对象属性

属性	描述
constructor	对创建该对象的函数的引用
length	字符串的长度
prototype	允许您向对象添加属性和方法

String 对象方法

方法	描述
anchor()	创建 HTML 锚。
big()	用大号字体显示字符串。
blink()	显示闪动字符串。
bold()	使用粗体显示字符串。
charAt()	返回在指定位置的字符。
charCodeAt()	返回在指定的位置的字符的 Unicode 编码。
concat()	连接字符串。

<code>fixed()</code>	以打字机文本显示字符串。
<code>fontcolor()</code>	使用指定的颜色来显示字符串。
<code>fontsize()</code>	使用指定的尺寸来显示字符串。
<code>fromCharCode()</code>	从字符编码创建一个字符串。
<code>indexOf()</code>	检索字符串。
<code>italics()</code>	使用斜体显示字符串。
<code>lastIndexOf()</code>	从后向前搜索字符串。
<code>link()</code>	将字符串显示为链接。
<code>localeCompare()</code>	用本地特定的顺序来比较两个字符串。
<code>match()</code>	找到一个或多个正在表达式的匹配。
<code>replace()</code>	替换与正则表达式匹配的子串。
<code>search()</code>	检索与正则表达式相匹配的值。
<code>slice()</code>	提取字符串的片断，并在新的字符串中返回被提取的部分。
<code>small()</code>	使用小字号来显示字符串。
<code>split()</code>	把字符串分割为字符串数组。
<code>strike()</code>	使用删除线来显示字符串。
<code>sub()</code>	把字符串显示为下标。
<code>substr()</code>	从起始索引号提取字符串中指定数目的字符。
<code>substring()</code>	提取字符串中两个指定的索引号之间的字符。
<code>sup()</code>	把字符串显示为上标。
<code>toLocaleLowerCase()</code>	把字符串转换为小写。
<code>toLocaleUpperCase()</code>	把字符串转换为大写。
<code>toLowerCase()</code>	把字符串转换为小写。
<code>toUpperCase()</code>	把字符串转换为大写。
<code>toSource()</code>	代表对象的源代码。
<code>toString()</code>	返回字符串。
<code>valueOf()</code>	返回某个字符串对象的原始值。

RegExp 对象属性

属性	描述
<code>global</code>	RegExp 对象是否具有标志 <code>g</code> 。
<code>ignoreCase</code>	RegExp 对象是否具有标志 <code>i</code> 。
<code>lastIndex</code>	一个整数，标示开始下一次匹配的字符位置。
<code>multiline</code>	RegExp 对象是否具有标志 <code>m</code> 。
<code>source</code>	正则表达式的源文本。

RegExp 对象方法

方法	描述
compile	编译正则表达式。
exec	检索字符串中指定的值。返回找到的值，并确定其位置。
test	检索字符串中指定的值。返回 true 或 false。

Browser 对象

对象	描述
Window	Window 对象表示浏览器中打开的窗口。
Navigator	Navigator 对象包含有关浏览器的信息。
Screen	Screen 对象包含有关客户端显示屏幕的信息。
History	History 对象包含用户（在浏览器窗口中）访问过的 URL。
Location	Location 对象包含有关当前 URL 的信息。

Window 对象属性

属性	描述
closed	返回窗口是否已被关闭。
defaultStatus	设置或返回窗口状态栏中的默认文本。
document	对 Document 对象的只读引用。请参阅 Document 对象。
history	对 History 对象的只读引用。请参阅 History 对象。
innerheight	返回窗口的文档显示区的高度。
innerwidth	返回窗口的文档显示区的宽度。
length	设置或返回窗口中的框架数量。
location	用于窗口或框架的 Location 对象。请参阅 Location 对象。
name	设置或返回窗口的名称。
Navigator	对 Navigator 对象的只读引用。请参阅 Navigator 对象。
opener	返回对创建此窗口的窗口的引用。
outerheight	返回窗口的外部高度。
outerwidth	返回窗口的外部宽度。
pageXOffset	设置或返回当前页面相对于窗口显示区左上角的 X 位置。
pageYOffset	设置或返回当前页面相对于窗口显示区左上角的 Y 位置。
parent	返回父窗口。
Screen	对 Screen 对象的只读引用。请参阅 Screen 对象。
self	返回对当前窗口的引用。等价于 Window 属性。

status	设置窗口状态栏的文本。
top	返回最顶层的先辈窗口。
window	window 属性等价于 self 属性，它包含了对窗口自身的引用。
screenLeft screenTop screenX screenY	只读整数。声明了窗口的左上角在屏幕上的 x 坐标和 y 坐标。 IE、Safari 和 Opera 支持 screenLeft 和 screenTop ，而 Firefox 和 Safari 支持 screenX 和 screenY 。

Window 对象方法

方法	描述
alert()	显示带有一段消息和一个确认按钮的警告框。
blur()	把键盘焦点从顶层窗口移开。
clearInterval()	取消由 setInterval() 设置的 timeout 。
clearTimeout()	取消由 setTimeout() 方法设置的 timeout 。
close()	关闭浏览器窗口。
confirm()	显示带有一段消息以及确认按钮和取消按钮的对话框。
createPopup()	创建一个 pop-up 窗口。
focus()	把键盘焦点给予一个窗口。
moveBy()	可相对窗口的当前坐标把它移动指定的像素。
moveTo()	把窗口的左上角移动到一个指定的坐标。
open()	打开一个新的浏览器窗口或查找一个已命名的窗口。
print()	打印当前窗口的内容。
prompt()	显示可提示用户输入的对话框。
resizeBy()	按照指定的像素调整窗口的大小。
resizeTo()	把窗口的大小调整到指定的宽度和高度。
scrollBy()	按照指定的像素值来滚动内容。
scrollTo()	把内容滚动到指定的坐标。
setInterval()	按照指定的周期（以毫秒计）来调用函数或计算表达式。
setTimeout()	在指定的毫秒数后调用函数或计算表达式。

Navigator 对象属性

属性	描述
appName	返回浏览器的代码名。
appMinorVersion	返回浏览器的次级版本。
appName	返回浏览器的名称。

appVersion	返回浏览器的平台和版本信息。
browserLanguage	返回当前浏览器的语言。
cookieEnabled	返回指明浏览器中是否启用 cookie 的布尔值。
cpuClass	返回浏览器系统的 CPU 等级。
onLine	返回指明系统是否处于脱机模式的布尔值。
platform	返回运行浏览器的操作系统平台。
systemLanguage	返回 OS 使用的默认语言。
userAgent	返回由客户机发送服务器的 user-agent 头部的值。
userLanguage	返回 OS 的自然语言设置。

Navigator 对象方法

方法	描述
javaEnabled()	规定浏览器是否启用 Java 。
taintEnabled()	规定浏览器是否启用数据污点 (data tainting)。

Screen 对象属性

属性	描述
availHeight	返回显示屏幕的高度 (除 Windows 任务栏之外)。
availWidth	返回显示屏幕的宽度 (除 Windows 任务栏之外)。
bufferDepth	设置或返回调色板的比特深度。
colorDepth	返回目标设备或缓冲器上的调色板的比特深度。
deviceXDPI	返回显示屏幕的每英寸水平点数。
deviceYDPI	返回显示屏幕的每英寸垂直点数。
fontSmoothingEnabled	返回用户是否在显示控制面板中启用了字体平滑。
height	返回显示屏幕的高度。
logicalXDPI	返回显示屏幕每英寸的水平方向的常规点数。
logicalYDPI	返回显示屏幕每英寸的垂直方向的常规点数。
pixelDepth	返回显示屏幕的颜色分辨率 (比特每像素)。
updateInterval	设置或返回屏幕的刷新率。
width	返回显示器屏幕的宽度。

History 对象属性

属性	描述
length	返回浏览器历史列表中的 URL 数量。

History 对象方法

方法	描述
back()	加载 history 列表中的前一个 URL。
forward()	加载 history 列表中的下一个 URL。
go()	加载 history 列表中的某个具体页面。

Location 对象属性

属性	描述
hash	设置或返回从井号 (#) 开始的 URL (锚)。
host	设置或返回主机名和当前 URL 的端口号。
hostname	设置或返回当前 URL 的主机名。
href	设置或返回完整的 URL。
pathname	设置或返回当前 URL 的路径部分。
port	设置或返回当前 URL 的端口号。
protocol	设置或返回当前 URL 的协议。
search	设置或返回从问号 (?) 开始的 URL (查询部分)。

Location 对象方法

属性	描述
assign()	加载新的文档。
reload()	重新加载当前文档。
replace()	用新的文档替换当前文档。

JavaScript 函数

函数	描述
decodeURI ()	解码某个编码的 URI。
decodeURIComponent ()	解码一个编码的 URI 组件。
encodeURI ()	把字符串编码为 URI。
encodeURIComponent ()	把字符串编码为 URI 组件。
escape ()	对字符串进行编码。
eval ()	计算 JavaScript 字符串，并把它作为脚本代码来执行。
getClass ()	返回一个 JavaScript 的 JavaClass。
isFinite ()	检查某个值是否为有穷大的数。
isNaN ()	检查某个值是否是数字。

Number ()	把对象的值转换为数字。
parseFloat ()	解析一个字符串并返回一个浮点数。
parseInt ()	解析一个字符串并返回一个整数。
String ()	把对象的值转换为字符串。
unescape ()	对由 escape () 编码的字符串进行解码。

JavaScript 常用事件

事件	描述
onClick	当用户单击鼠标按钮时，产生的事件
onDbClick	当用户双击鼠标按钮时，产生的事件
onChange	当文本框的内容改变的时候，发生的事件
onFocus	当光标落在文本框中的时候，发生的事件
onLoad	当当前的网页被显示的时候，发生的事件
onUnLoad	当当前的网页被关闭的时候，发生的事件
onBlur	当光标离开文本框中的时候，发生的事件
onMouseOver	当鼠标移动到页面元素上方时发生的事件
onMouseOut	当鼠标离开页面元素上方时发生的事件
onAbort	当页面上图像没完全下载时，访问者单击浏览器上停止按钮的事件
onAfterUpdate	页面特定数据元素完成更新的事件
onBeforeUpdate	页面特定数据元素被改变且失去焦点的事件
onBounce	移动的 Marquee 文字到达移动区域边界的事件
onError	页面或页面图像下载出错事件
onFinish	移动的 Marquee 文字完成一次移动的事件
onHelp	访问者单击浏览器上帮助按钮的事件
onKeyDown	访问者按下键盘一个或几个键的事件
OnKeyPress	访问者按下键盘一个或几个键后且松开的事件
onKeyUp	访问者按下键盘一个或几个键后松开的事件
onMouseDown	访问者按下鼠标按钮的事件
onMouseMove	访问者鼠标在某页面元素范围内移动的事件
onMouseUp	访问者松开鼠标按钮的事件
onMove	窗口或窗框被移动的事件
onReadyStateChange	特定页面元素状态被改变的事件
onReset	页面上表单元素的值被重置的事件
onResize	访问者改变窗口或窗框大小的事件
onScroll	访问者使用滚动条的事件
onStart	Marquee 文字开始移动的事件
onSubmit	页面上表单被提交的事件