

符号	解释	属性
<code>var obj=/ /</code>	创建正则对象；即赋值被注释掉，及运行被结束	
/正则表达式主体/修饰符	可选属性	
<code>i</code>	执行对大小写不敏感的匹配；	修饰符
<code>g</code>	执行全局匹配 查找所有匹配而非在找到第一个匹配后停止	修饰符
<code>m</code>	执行多行匹配	修饰符
<code>y</code>	全局匹配。ES6新增	修饰符
<code>[abc]</code>	查找方括号之间的任何字符。	
<code>[0-9]</code>	查找任何从 0 至 9 的数字	
<code>()</code>	查找 () 中内容	
<code>(x\ y)</code>	查找任何以 \ 分隔的选项	
<code>\d</code>	查找数字	
<code>\D</code>	匹配一个非数字字符	
<code>\s</code>	查找空白字符	
<code>\b</code>	匹配单词边界	
<code>\w</code>	匹配字母数字或下划线	
<code>\W</code>	表示除了字母、数字下划线的字符	
<code>\uxxxx</code>	查找以十六进制数 xxxx 规定的 Unicode 字符	
<code>/^/</code>	匹配输入的开始	
<code>/[^]/</code>	表示非	
<code>\$</code>	匹配输入的结束。	
<code>*</code>	匹配前一个表达式 0 次或多次。等价于 {0, }	
<code>+</code>	匹配前面一个表达式 1 次或者多次。等价于 {1, }	
<code>?</code>	非贪心的匹配前面一个表达式 0 次或者 1 次。等价于 {0, 1}	

符号	解释	属性
.	默认匹配除换行符之外的任何单个字符。	
{ }	个数	
\1	对第一个捕获组的引用	
\$1	第一个分组的值	

• 新建

- `var regex = /xyz/g;`
- `var regex = new RegExp('xyz', 'i');`

- **test()**：正则实例对象的 `test` 方法返回一个布尔值，表示当前模式是否能匹配参数字符串。
`/cat/.test('cats and dogs') // true`
- **exec()**：正则实例对象的 `exec` 方法，用来返回匹配结果。如果发现匹配，就返回一个数组，成员是匹配成功的子字符串，否则返回 `null`。
- **match()**：字符串实例对象的 `match` 方法对字符串进行正则匹配，返回匹配结果。
- **search()**：字符串对象的 `search` 方法，返回第一个满足条件的匹配结果在整个字符串中的位置。如果没有任何匹配，则返回 `-1`。
`'_x_x'.search(/x/) // 1`
- **replace()**：字符串对象的 `replace` 方法可以替换匹配的值。它接受两个参数，第一个是正则表达式，表示搜索模式，第二个是替换的内容。正则表达式如果不加 `g` 修饰符，就替换第一个匹配成功的值，否则替换所有匹配成功的值。
`str.replace(search, replacement);`
- **split()**：字符串对象的 `split` 方法按照正则规则分割字符串，返回一个由分割后的各个部分组成的数组。
`str.split(separator, [limit])`