

JavaScript - Objetos predefinidos - String

Es un objeto predefinido de JavaScript para manejo de cadenas de caracteres.

Propiedades del objeto String

`length` longitud de la cadena de caracteres asociada al objeto

Se crea con el constructor `String`, pero cualquier variable que contiene una cadena de texto puede ser tratada como `String` (JavaScript se encarga de hacer las conversiones necesarias):

```
cadena1 = new String ("Un String");
longitud = cadena1.length;
cadena2 = "Una cadena de caracteres";
longitud = cadena2.length;
// también se puede poner
longitud="Una cadena de caracteres".length
```

Métodos del objeto String

`anchor(nombreDelAncla)`

crea un anclaje HTML:

```
referencia = new String ("Zona inferior");
document.write (referencia.anchor ("inferior"));
```

Sería equivalente a:

` Zona inferior `

`big()`

Devuelve una cadena que representa el texto en formato BIG de HTML.

`blink()`

Crea el efecto de una cadena parpadeante.

`bold()`

Devuelve una cadena que representa el texto en negrita (`...`) en HTML.

`charAt(posición)`

Devuelve el caracter correspondiente a la posición indicada (teniendo en cuenta que las cadenas comienzan con índice cero).

`charCodeAt(caracter)`

Devuelve el código (codificación ISO-Latin-1) correspondiente al carácter situado en la posición indicada (las cadenas comienzan con índice cero).

`concat(cadena1,...,cadenan)`

Concatena dos o más cadenas y devuelve como resultado esa unión.

`fixed()`

Devuelve una cadena que representa el texto en formato TT con fuente monoespaciada de HTML.

`fontcolor(color)`

Modifica el color de la fuente de la cadena con el color pasado como parametro.

<code>fontsize(número)</code>	Modifica el tamaño de la fuente de una cadena con el valor pasado como parametro.
<code>fromCharCode(cod1,...,codn)</code>	Convierte valores Unicode en caracteres.
<code>indexOf(valor[,inicio])</code>	Devuelve el índice de la primera aparición del valor especificado dentro de la cadena desde la posición inicio, si se omite inicio empieza a buscar desde el primer carácter empezando desde la izquierda de la cadena. Si no se encuentra devuelve -1:
	<pre>cadena = "uno dos tres cuatro"; posicionDeDos = cadena.indexOf ("dos"); // posicionDeDos = 4</pre>
<code>italics()</code>	Devuelve una cadena que representa el texto en formato <I>...</I> (cursiva) de HTML.
<code>lastIndexOf(valor[,fin])</code>	Devuelve el índice de la primera aparición del valor especificado dentro de la cadena desde la posición fin, si se omite inicio empieza a buscar desde el primer carácter empezando desde la derecha de la cadena. Si no se encuentra devuelve -1:
	<pre>cadena = "uno dos tres cuatro dos"; posicionDeDos = cadena.lastIndexOf ("dos"); // posicionDeDos = 20</pre>
<code>link(href)</code>	Construye una cadena que se corresponde con la marca de enlace (<a>) de HTML:
	<pre>enlace = new String ("Epsilon Eridani"); marcaEnlace = enlace.link ("http://www.epsilon-eridani.com"); // marcaEnlace = Epsilon Eridani</pre>
<code>match(patrón)</code>	Busca una coincidencia en una cadena y devuelve todas las coincidencias encontradas en un array.
<code>replace(patrón, nuevaCadena)</code>	Busca una coincidencia en una cadena y si existe, la reemplaza por otra cadena pasada como parámetro.
<code>search(patrón)</code>	Busca una coincidencia en una cadena y devuelve la posición de la coincidencia o -1 si no existe ninguna.

`slice(inicio, ultimo)`

Extrae un trozo de una cadena desde inicio y último sin incluirlo, ambos de índice cero, o el final de la cadena si se omite este segundo:

```
cadena = "dos trozos";  
trozo = cadena.slice (4,9); // trozo='trozos'
```

Si el segundo carácter es negativo, indica la posición comenzando por la parte derecha de la cadena original:

```
trozo = cadena.slice (4,-1); // trozo='trozo'
```

`small()`

Devuelve una cadena que se corresponde con el formato SMALL de HTML.

`split(separador[,limite])`

Divide una cadena en trozos y los devuelve en un array. Se pasa como parámetro la cadena utilizada como separador:

```
cadena = "uno dos tres cuatro dos";  
trozos = cadena.split (" "); // el separador es el  
espacio  
document.write (trozos[0]); // trozos[0] = 'uno'
```

Se puede especificar un segundo parámetro que indica el número máximo de trozos:

```
trozos = cadena.split (" ",2);  
// se crea un array de 2 elementos
```

`strike()`

Muestra una cadena tachada.

`sub()`

Devuelve una cadena con el formato SUB (subíndice) de HTML.

`substr(inicio[,cantidad])`

Devuelve una subcadena a partir de los parámetros 'inicio' y 'longitud':

```
cadena = "uno dos tres cuatro dos";  
subcadena = cadena.substr (4, 3); // subcadena = 'dos'
```

Si no se especifica la longitud, se considera hasta el final de la cadena. Si 'inicio' es un valor negativo se cuenta la posición inicial a partir del final de la cadena.

`substring(indiceA[,indiceB])` Devuelve una subcadena que está comprendida entre dos índices de la cadena actual, sin incluir el carácter del índiceB, ambos de índice cero:

```
cadena = "uno dos tres cuatro dos";  
subcadena = cadena.substring (4, 7); // subcadena =  
'dos'
```

Si no se especifica índiceB, se considera hasta el final de la cadena.

`sup()` Devuelve una cadena con el formato SUP (superíndice) de HTML.

`toLowerCase()` Devuelve una cadena con los caracteres convertidos a minúsculas.

`toUpperCase()` Devuelve una cadena con los caracteres convertidos a mayúsculas.

`trim()`
`trimLeft()`
`trimRight()` Eliminan los espacios al principio y al final de la cadena, al principio solamente o al final solamente, respectivamente.