JavaScript String Functions, Operators and Properties

An easy reference for all JavaScript functions and operations on strings HTML-5.com is a great guide for web developers. TV Series & Actors and Actresses. Follow TV Series and HTML 5 on Google+.

HTML-5.com ➢ itemscopehttp://data-vocabulary.org/BreadcrumbHTML 5 ➢ itemscopehttp://data-vocabulary.org/BreadcrumbJavaScript Code ➢ itemscopehttp://data-vocabulary.org/ BreadcrumbJavaScript DOM ➢ itemscopehttp://data-vocabulary.org/Breadcrumb

JavaScript String Functions, Operators and Properties

JavaScript String Functions, Operators and Properties

```
<script type="text/javascript"> var str = "HTML tags start with "<" & end with ">".";
window.addEventListener("load", function(ev) { document.getElementById("string-
length").textContent = str.length; document.getElementById("string-plus").textContent
= "This is" + " a test."; document.getElementById("string-plus-equals").textContent
= str + " Right on!"; // document.getElementById("string-bold").innerHTML =
htmlEncode(str).bold(); document.getElementById("string-charat").textContent
= str.charAt(2); document.getElementById("string-charcodeat").textContent =
str.charCodeAt(2); document.getElementById("string-concat").textContent = str.concat(" ",
"Right on!"); document.getElementById("string-indexof").textContent = str.indexOf("with");
document.getElementById("string-indexof-position").textContent = str.indexOf("with",
20); document.getElementById("string-lastindexof").textContent = str.lastIndexOf("with");
document.getElementById("string-lastindexof-position").textContent = str.lastIndexOf("with",
20); document.getElementById("string-localecompare-aa").textContent =
"A".localeCompare("a"); document.getElementById("string-localecompare-ab1").textContent
= "A".localeCompare("b"); document.getElementById("string-localecompare-
ab2").textContent = "a".localeCompare("B"); document.getElementById("string-
match").textContent = str.match(/with/gi); document.getElementById("string-
replace").textContent = str.replace("start", "begin"); document.getElementById("string-
search").textContent = str.search(/[^a-z\s]/i); document.getElementById("string-
slice").textContent = str.slice(16, 31); document.getElementById("string-
split").textContent = str.split(" ")[2]; document.getElementById("string-split-
regexp").textContent = str.split(/\s/)[3]; document.getElementById("string-
split-limit").textContent = "first " + str.split(" ", 5).length + " of " + str.split("
").length; document.getElementById("string-tolocalelowercase").textContent =
str.toLocaleLowerCase(); document.getElementById("string-tolocaleuppercase").textContent
= str.toLocaleUpperCase(); document.getElementById("string-tolowercase").textContent
= str.toLowerCase(); document.getElementById("string-tostring").textContent
= str.toString(); document.getElementById("string-touppercase").textContent
= str.toUpperCase(); document.getElementById("string-trim").textContent =
" TEST ".trim(); document.getElementById("string-valueof").textContent =
str.valueOf(); document.getElementById("string-from-char-code-dec").textContent
= String.fromCharCode(9733, 10003); document.getElementById("string-
from-char-code-hex").textContent = String.fromCharCode(0x2605, 0x2713);
document.getElementById("string-decodeuri").textContent = decodeURI("/The
%20Firm/"); document.getElementById("string-decodeuricomponent").textContent
```

= decodeURIComponent("decimal%E2%80%94To%E2%80%94%25") + "=" + decodeURIComponent("pct%20%3D%20dec%23%20*%20100"); var elem = document.createElement("a"); elem.href = encodeURI("http://TVSeries.com/The Firm/"); elem.textContent = elem.href; document.getElementById("string-encodeuri").appendChild(elem); document.getElementById("string-encodeuricomponent").textContent = encodeURIComponent("decimal—To—%") + "=" + encodeURIComponent("pct = dec# * 100"); document.getElementById("string-escape").textContent = escape("decimal—To—%") + "=" + escape("pct = dec# * 100"); document.getElementById("string-eval").textContent = eval(".67 * 100") + "%"; document.getElementById("string-parsefloat").textContent = parseFloat("123.45"); document.getElementById("string-parseint").textContent = parseInt("123"); document.getElementById("string-json-parse").textContent = "typeof obj = " + (typeof JSON.parse('{"foo":"bar"}')); document.getElementById("string-json-stringify").textContent = JSON.stringify({foo:"bar"}); }); var htmlEntities = [">", "&", "'", """, "<"]; function htmlEncode(str) { return str.replace(/[>&'"<]/g, function(str, offset, s) { return htmlEntities[str.charCodeAt(0) % 25 % 6]; }); } </script>

String used in examples below
`var str = "HTML tags start with “<” & end with “>”.";`

The results in the examples below are from actual *working* examples of JavaScript, resulting from running actual JavaScript code in your browser, rather than simply the expected results put in HTML code. For example, a <dfn>locale-sensitive comparison</dfn> as performed by localeCompare will be based on the locale in *your* browser.

String properties

	Result	String property	Use / Result
get only	Number	<var>string</var>.**length**	get length of String
Example:	`str.length`		

String operators

Result	String operator	Use / Result
String	**=** <var>value</var>	String assignment operator (JavaScript equals)
Example:	`var newstr = "value"`	a String with a value of "value"
String	<var>string1</var> **+** <var>string2</var>	concatenate two Strings
Example:	`var newstr = "This is" + " a test.";`	
String	**+=** <var>string2</var>	append a String to another String
Example:	`str += " Right on!";`	

Static methods of String global object

Static Method	Use / Result	
String	**new String**(<var>value</var>)	String constructor
Examples:	`var newstr = new String("value")`	a String with a value of "value"
`var newstr = new String()`	an empty String	
String	**String**(<var>object</var>)	cast <var>object</var> to String
Examples:	`String(obj)`	
String	String.**fromCharCode**(<var>char</var>[, ... <var>charN</var>])	create a String from character codes
Examples:	`var newstr = String.fromCharCode(9733, 10003)`	
`var newstr = String.fromCharCode(0x2605, 0x2713)`		

String functions

Result	String function	Use / Result
String	<var>string</var>.**charAt**(<var>position</var>)	get character at position in string
Example:	`str.charAt(2)`	
Number	<var>string</var>.**charCodeAt**(<var>position</var>)	get character code at position in string
Example:	`str.charCodeAt(2)`	
String	<var>string</var>.**concat**(<var>string1</var>[, ... <var>stringN</var>])	append one or more Strings
Example:	`str.concat(" ", "Right on!")`	
Number	<var>string</var>.**indexOf**(<var>searchString</var>[, <var>position</var>])	find position of <var>searchString</var> within string; you can also search for a regular expression

		using String.search(/<var>regexp</var>/<var>flags</var>)
Examples:	`str.indexOf("with")`	
`str.indexOf("with", 16)`		
Number	<var>string</var>.**lastIndexOf**(<var>searchString</var>[, <var>position</var>])	find last position of <var>searchString</var> within string
Examples:	`str.lastindexOf("with")`	
`str.lastindexOf("with", 31)`		
Number	<var>string</var>.**localeCompare**(<var>string</var>, <var>string2</var>)	locale-sensitive comparison of one string with another string Note that: • comparison is dependent on the *client* system's locale, *not* the server-side system's locale, and • although a case-sensitive comparision is used, which letter in the alphabet is considered a higher priority for determining order than its case - in English, for example, lower case comes before upper case, so the order of alphabetic characters is "a", "A", "b", "B", etc. (see examples below)
Example:	`"A".localeCompare("a")`	
`"A".localeCompare("b")`		
`"a".localeCompare("B")`		
Array	<var>string</var>.**match**(/<var>regexp</var>/[<var>flags</var>])	retrieve substrings of a String that match a given regular expression
Example:	`str.match(/with/gi)`	
String	<var>string</var>.**replace**(<var>searchString</var>	replace part of one string with another string

	var>, <var>replacement</var>) <var>string</var>.**replace**(/<var>regexp</var>/<var>flags</var>, <var>replacement</var>)	
Example:	`str.replace("start", "begin")`	
Number	<var>string</var>.**search**(/<var>regexp</var>/[<var>flags</var>])	get index of a substring that matches a regular expression; to match a string instead use String.indexOf(<var>substring</var>[, <var>position</var>])
Example:	`str.search(/[^a-z\s]/i)`	
String	<var>string</var>.**slice**(<var>start</var>, <var>end</var>)	get a substring of a string
Example:	`str.slice(16, 31)`	
Array	<var>string</var>.**split**(<var>separator</var>[, <var>limit</var>]) <var>string</var>.**split**(/<var>regexp</var>/[, <var>limit</var>])	split String based on <var>separator</var>
Examples:	`str.split(" ")[2] // third word`	
`str.split(/\s/)[3] // fourth word`		
`"first " + str.split(" ", 5).length + " of " + str.split(" ").length`		
String	<var>string</var>.**toLocaleLowerCase**()	convert string to lower case based on client locale
Example:	`str.toLocaleLowerCase()`	
String	<var>string</var>.**toLocaleUpperCase**()	convert string to upper case based on client locale
Example:	`str.toLocaleUpperCase()`	
String	<var>string</var>.**toLowerCase**()	convert string to lower case

Example:	`str.toLowerCase()`	
String	\<var>string\</var>.**toString**()	the String itself, same as String.valueOf()
Example:	`str.toString()`	
String	\<var>string\</var>.**toUpperCase**()	convert string to upper case
Example:	`str.toUpperCase()`	
String	\<var>string\</var>.**trim**()	remove leading and trailing spaces
Example:	`" TEST ".trim()`	
String	\<var>string\</var>.**valueOf**()	the String itself, same as String.toString()
Example:	`str.valueOf()`	

Global functions with String parameters

For the difference between encodeURI, encodeURIComponent and escape and information on the specific characters used for encoding, see JavaScript percent encoding functions.

Function / Operator	Use / Result	
String	**decodeURI**(\<var>encodedURI\</var>)	decode an encoded URI
Examples:	`decodeURI("/The %20Firm/")`	
String	**decodeURIComponent**(\<var>encodedComponent\</var>)	decoded query parameter or other encoded URI component
Examples:	`decodeURIComponent("decimal %E2%80%94To %E2%80%94%25")` `+ "=" +` `decodeURIComponent("pct %20%3D%20dec%23%20* %20100")`	
String	**encodeURI**(\<var>unencodedURI\</var>)	encode a URI
Examples:	`document.createElement("a").href = encodeURI("http:// TVSeries.com/The Firm/");`	
String	**encodeURIComponent**(\<var>unencodedComponent\</var>)	encoded query string parameter or other URI component

Examples:	`encodeURIComponent("decimal—To—%") + "=" + encodeURIComponent("pct = dec# * 100")`	
String	**escape**(<var>string</var>)	*deprecated* escape all special characters in a string
Examples:	`escape("decimal—To—%") + "=" + escape("pct = dec# * 100")`	
(any)	**eval**(<var>string1</var>[, ... <var>stringN</var>])	evaluate <var>string</var> as a JavaScript expression, returning the value of the last expression
Examples:	`eval(".67 * 100") + "%"`	
Number	**parseFloat**(<var>string</var>)	convert String to a floating point Number
Examples:	`var intval = parseFloat("123.45");`	
Number	**parseInt**(<var>string</var>)	convert String to an integer Number
Examples:	`var intval = parseInt("123");`	

JSON functions

Result	JSON function	Result
Object	**JSON.parse**(<var>string</var>)	convert JSON string to JavaScript object
Example:	`var obj = JSON.parse( '{"foo":"bar"}' )`	
String	**JSON.stringify**(<var>object</var>)	convert JavaScript object to JSON string
Example:	`var newstr = JSON.stringify( { foo: "bar" } );`	

[back to top](#)

THE END