

SYNTAX

```
'This is a string'
"This is also a string"
String(another string)
```

PROPERTIES

```
var s1 = "hello";
s1.length;    // returns 5
```

ARRAYS and Strings

Split string into array by specified separator:

```
var mlist = "first,second,third";
mlist.split(",");
// returns ["first","second","third"]
mlist.split("i");
// returns ["f","rst,second,th","rd"]
```

Join array into string with specified separator:

```
var a = ["first","second","third"];
a.join(" - ");
// returns "first - second - third"
```

CONCATENATION

```
var s1 = "hello ";
var s2 = "world";
var s3 = s1+s2;
var s4 = s1.concat(s2);
// both s3 and s4 are "hello world"
```

ESCAPE Notation

| | |
|------------------------|-------------------|
| <code>\0</code> | NULL character |
| <code>\'</code> | single quote |
| <code>\"</code> | double quote |
| <code>\\</code> | backslash |
| <code>\n</code> | newline |
| <code>\r</code> | carriage return |
| <code>\v</code> | vertical tab |
| <code>\t</code> | tab |
| <code>\b</code> | backspace |
| <code>\f</code> | form feed |
| <code>\uXXXX</code> | unicode codepoint |
| <code>\u{X...X}</code> | unicode codepoint |

CHARACTER Codes

Build string from character code:

```
var m = String.fromCharCode(72,69,76,76,79);
// m = "HELLO";
```

Get character code at position:

```
var m = "HELLO";
m.charCodeAt(2);
// returns 76
```

CHARACTERS at Index

Cannot be used to (re)define a character in a position

```
var s1 = "hello";
s1.charAt(2);    // returns "l"
s1[1];          // returns "e"
```

SUBSTRINGS

Get first instance of substring:

```
var lorem = "Lorem ipsum";
lorem.indexOf("m");
// returns 4
lorem.indexOf("ip");
// returns 6
```

Get last instance of substring:

```
var lorem = "Lorem ipsum";
lorem.lastIndexOf("m");
// returns 10
lorem.lastIndexOf("ip");
// returns 6
```

Get substring by start and end indices:

```
var lorem = "Lorem ipsum dolor sit amet";
lorem.slice( 6, 17 );
// returns "ipsum dolor"
```

Get substring by start index and length:

```
var lorem = "Lorem ipsum dolor sit amet";
lorem.substr( 6, 11 );
// returns "ipsum dolor"
```

Test if substring appears:

```
var lorem = "Lorem ipsum dolor sit amet";
lorem.includes("ipsum dolor");
// returns true
```

HTML

Create and display a link:

```
var ltext = "Google it";
var url = "http://google.com";
document.write("Just " + ltext.link(url));
```

Create and display an anchor:

```
var aText = "Section One";
document.write(aText.anchor("section1"));
```

FORMATTING

Convert to lower case:

```
var myString = "Hello WORLD";
myString.toLowerCase();
// returns "hello world"
```

Convert to upper case:

```
var myString = "Hello WORLD";
myString.toUpperCase();
// returns "HELLO WORLD"
```

Remove whitespace before, after, or both:

```
var myString = " Hello world ";
myString.trim();
// returns "Hello world"
myString.trimLeft();
// returns "Hello world "
myString.trimRight();
// returns " Hello world"
```

Formatting numbers with options:

```
var n = 50000;
n.toLocaleString(); //return "50,000"
n.toLocaleString('en-US',
  { style: 'currency', currency: 'USD' });
// returns "$50,000.00"
```

NUMBERS

Convert to first integer, if possible, with optional base:

```
parseInt("23 10");    // returns 23
parseInt("5 years");  // returns 5
parseInt("10",8);     // returns 8
```