

# String

## Property Summary

**Number** `#length`  
Gives the length of the string.

## Method Summary

String	<a href="#">#anchor</a> (nameAttribute) returns a copy of the string embedded within an anchor <A> tag set.
String	<a href="#">#big</a> () returns a copy of the string embedded within an <BIG> tag set.
String	<a href="#">#blink</a> () returns a copy of the string embedded within an <BLINK> tag set.
String	<a href="#">#bold</a> () returns a copy of the string embedded within an <B> tag set.
Number	<a href="#">#charAt</a> (index) returns a character of the string.
Number	<a href="#">#charCodeAt</a> (index) returns a decimal code of the char in the string.
String	<a href="#">#concat</a> (string2) returns a string that appends the parameter string to the string.
String	<a href="#">#concat</a> (string2, stringN) returns a string that appends the parameter string to the string.
Boolean	<a href="#">#equals</a> (other) returns a boolean that checks if the given string is equal to the string
Boolean	<a href="#">#equalsIgnoreCase</a> (other) returns a boolean that checks if the given string is equal to the string ignoring case
String	<a href="#">#fixed</a> () returns a copy of the string embedded within an anchor <TT> tag set.
String	<a href="#">#fontcolor</a> (color) returns a copy of the string embedded within an <FONT> tag set, the color param is assigned the the color attribute.
String	<a href="#">#fontsize</a> (size) returns a copy of the string embedded within an <FONT> tag set, The size param is set to the SIZE attribute
Number	<a href="#">#indexOf</a> (searchValue, fromIndex) returns the found index of the given string in string.
String	<a href="#">#italics</a> () returns a copy of the string embedded within an <I> tag set
Number	<a href="#">#lastIndexOf</a> (searchValue, fromIndex) returns the found index of the given string in string from the end.
String	<a href="#">#link</a> (hrefAttribute) returns a copy of the string embedded within an <A> tag set.
Number	<a href="#">#localeCompare</a> (otherString)
Array	<a href="#">#match</a> (regex) returns an array of strings within the current string that matches the regex.
String	<a href="#">#replace</a> (regex, newSubStr) returns a strings where all matches of the regex are replaced.
Number	<a href="#">#search</a> (regex) returns a index where the first match is found of the regex
String	<a href="#">#slice</a> (beginSlice) returns a substring of the string.
String	<a href="#">#slice</a> (beginSlice, endSlice) returns a substring of the string.
String	<a href="#">#small</a> () returns a copy of the string embedded within an <SMALL> tag set.
String	<a href="#">#split</a> (separator, limit) returns an array of objects whose elements are segments of the current string.
String	<a href="#">#strike</a> () returns a copy of the string embedded within an <STRIKE> tag set.
String	<a href="#">#sub</a> () returns a copy of the string embedded within an <SUB> tag set.
String	<a href="#">#substr</a> (start) returns a substring of the string from the start with the number of chars specified.
String	<a href="#">#substr</a> (start, length) returns a substring of the string from the start with the number of chars specified.
String	<a href="#">#substring</a> (indexA) Returns a substring of the string from the start index until the end index.
String	<a href="#">#substring</a> (indexA, indexB) Returns a substring of the string from the start index until the end index.
String	<a href="#">#sup</a> () returns a copy of the string embedded within an <SUP> tag set.
String	<a href="#">#toLocaleLowerCase</a> ()
String	<a href="#">#toLocaleUpperCase</a> ()
String	<a href="#">#toLowerCase</a> () returns a string with all lowercase letters of the current string.
String	<a href="#">#toUpperCase</a> () returns a string with all uppercase letters of the current string.

## Property Details

length

Gives the length of the string.

### Returns

Number

### Sample

```
string.length;
```

## Method Details

anchor

[String](#) **anchor**(nameAttribute)

returns a copy of the string embedded within an anchor <A> tag set.

### Parameters

{[String](#)} nameAttribute

### Returns

[String](#)

### Sample

```
string.anchor();
```

big

[String](#) **big**()

returns a copy of the string embedded within an <BIG> tag set.

### Returns

[String](#)

### Sample

```
string.big();
```

blink

[String](#) **blink**()

returns a copy of the string embedded within an <BLINK> tag set.

### Returns

[String](#)

### Sample

```
string.blink();
```

bold

[String](#) **bold**()

returns a copy of the string embedded within an <B> tag set.

### Returns

[String](#)

### Sample

```
string.bold();
```

charAt

[Number](#) **charAt**(index)

returns a character of the string.

### Parameters

{[Number](#)} index

### Returns

Number

### Sample

```
string.charAt(integer_position);
```

### charCodeAt

**Number** **charCodeAt**(index)

returns a decimal code of the char in the string.

#### Parameters

{**Number**} index

#### Returns

**Number**

### Sample

```
string.charCodeAt(integer_position);
```

### concat

**String** **concat**(string2)

returns a string that appends the parameter string to the string.

#### Parameters

{**String**} string2

#### Returns

**String**

### Sample

```
string.concat(string);
```

### concat

**String** **concat**(string2, stringN)

returns a string that appends the parameter string to the string.

#### Parameters

{**String**} string2

{**String**} stringN

#### Returns

**String**

### Sample

```
string.concat(string);
```

### equals

**Boolean** **equals**(other)

returns a boolean that checks if the given string is equal to the string

#### Parameters

{**String**} other

#### Returns

**Boolean**

### Sample

```
string.equals(string);
```

### equalsIgnoreCase

**Boolean** **equalsIgnoreCase**(other)

returns a boolean that checks if the given string is equal to the string ignoring case

#### Parameters

{**String**} other

#### Returns

**Boolean**

### Sample

```
string.equalsIgnoreCase(string);
```

fixed

[String](#) **fixed()**

returns a copy of the string embedded within an anchor <TT> tag set.

**Returns**

[String](#)

**Sample**

```
string.fixed();
```

fontcolor

[String](#) **fontcolor**(color)

returns a copy of the string embedded within an <FONT> tag set, the color param is assigned the the color attribute.

**Parameters**

[{String}](#) color

**Returns**

[String](#)

**Sample**

```
string.fontcolor(color);
```

fontsize

[String](#) **fontsize**(size)

returns a copy of the string embedded within an <FONT> tag set, The size param is set to the SIZE attribute

**Parameters**

[{Number}](#) size

**Returns**

[String](#)

**Sample**

```
string.fontsize(size);
```

indexOf

[Number](#) **indexOf**(searchValue, fromIndex)

returns the found index of the given string in string.

**Parameters**

[{String}](#) searchValue

[{Number}](#) fromIndex

**Returns**

[Number](#)

**Sample**

```
string.indexOf(string, startPosition);
```

italics

[String](#) **italics()**

returns a copy of the string embedded within an <I> tag set

**Returns**

[String](#)

**Sample**

```
string.italics();
```

lastIndexOf

[Number](#) **lastIndexOf**(searchValue, fromIndex)

returns the found index of the given string in string from the end.

**Parameters**

[{String}](#) searchValue

[{Number}](#) fromIndex

**Returns**

[Number](#)

### Sample

```
string.lastIndexOf(string, startPosition);
```

### link

**String** **link**(hrefAttribute)

returns a copy of the string embedded within an <A> tag set.

#### Parameters

{String} hrefAttribute

#### Returns

String

### Sample

```
string.link(url);
```

### localeCompare

**Number** **localeCompare**(otherString)

Replace with description

#### Parameters

{String} otherString

#### Returns

Number

### Sample

```
var s = "Have a nice day!";  
application.output(s.localeCompare("Hello"));
```

### match

**Array** **match**(regexp)

returns an array of strings within the current string that matches the regexp.

#### Parameters

{RegExp} regexp

#### Returns

Array

### Sample

```
string.match(regexpr);
```

### replace

**String** **replace**(regexp, newSubStr)

returns a strings where all matches of the regexp are replaced.

#### Parameters

{RegExp} regexp

{String} newSubStr

#### Returns

String

### Sample

```
string.replace(regexpr, replacestring);
```

### search

**Number** **search**(regexp)

returns a index where the first match is found of the regexp

#### Parameters

{RegExp} regexp

#### Returns

Number

### Sample

```
string.search(regexpr);
```

slice

**String slice**(beginSlice)

returns a substring of the string.

#### Parameters

{**Number**} beginSlice

#### Returns

**String**

### Sample

```
string.slice(start,end);
```

slice

**String slice**(beginSlice, endSlice)

returns a substring of the string.

#### Parameters

{**Number**} beginSlice

{**Number**} endSlice

#### Returns

**String**

### Sample

```
string.slice(start,end);
```

small

**String small**()

returns a copy of the string embedded within an <SMALL> tag set.

#### Returns

**String**

### Sample

```
string.small();
```

split

**String split**(separator, limit)

returns an array of objects whose elements are segments of the current string.

#### Parameters

{**String**} separator

{**Number**} limit

#### Returns

**String**

### Sample

```
string.split(delimiter,limitInteger);
```

strike

**String strike**()

returns a copy of the string embedded within an <STRIKE> tag set.

#### Returns

**String**

### Sample

```
string.strike();
```

sub

**String sub**()

returns a copy of the string embedded within an <SUB> tag set.

**Returns**

[String](#)

**Sample**

```
string.sub();
```

substr

[String](#) **substr**(start)

returns a substring of the string from the start with the number of chars specified.

**Parameters**

{[Number](#)} start

**Returns**

[String](#)

**Sample**

```
string.substr(start, number_of_chars);
```

substr

[String](#) **substr**(start, length)

returns a substring of the string from the start with the number of chars specified.

**Parameters**

{[Number](#)} start

{[Number](#)} length

**Returns**

[String](#)

**Sample**

```
string.substr(start, number_of_chars);
```

substring

[String](#) **substring**(indexA)

Returns a substring of the string from the start index until the end index.

**Parameters**

{[Number](#)} indexA

**Returns**

[String](#)

**Sample**

```
string.substring(start, end);
```

substring

[String](#) **substring**(indexA, indexB)

Returns a substring of the string from the start index until the end index.

**Parameters**

{[Number](#)} indexA

{[Number](#)} indexB

**Returns**

[String](#)

**Sample**

```
string.substring(start, end);
```

sup

[String](#) **sup**()

returns a copy of the string embedded within an <SUP> tag set.

**Returns**

[String](#)



### Sample

```
string.sup();
```

toLocaleLowerCase

[String](#) **toLocaleLowerCase()**

Replace with description

**Returns**

[String](#)

### Sample

```
var s = "Have a nice day!";
application.output(s.toLocaleLowerCase());
```

toLocaleUpperCase

[String](#) **toLocaleUpperCase()**

Replace with description

**Returns**

[String](#)

### Sample

```
var s = "Have a nice day!";
application.output(s.toLocaleUpperCase());
```

toLowerCase

[String](#) **toLowerCase()**

returns a string with all lowercase letters of the current string.

**Returns**

[String](#)

### Sample

```
string.toLowerCase();
```

toUpperCase

[String](#) **toUpperCase()**

returns a string with all uppercase letters of the current string.

**Returns**

[String](#)

### Sample

```
string.toUpperCase();
```