

JavaScript String Methods

- description : JavaScript String Methods
- author :
- email : shlim@repia.com
- lastupdate : 2021-04-23

The source of this article

[JavaScript String Methods](#)

String Methods and Properties

“John Doe” (Primitive values) 가 (가).

JavaScript ,

String Length

length

```

<!DOCTYPE html>
<html>
<body>
  <h2>JavaScript String Properties</h2>
  <p>The length property returns the length of a string</p>
  <p id="demo"></p>
  <script>
    let txt = "ABCDEFGHIJKLMNOPQRSTUVWXYZ";
    let sln = txt.length;
    document.getElementById("demo").innerHTML = sln;
  </script>
</body>
</html>

```

Finding a String in a String

`indexOf()` _____(occurrence) ()

```
<!DOCTYPE html>
<html>
<body>
  <h2>JavaScript String Methods</h2>
  <p>The indexOf() method returns the position of the first occurrence of a
specified text:</p>
  <p id="demo"></p>
  <script>
    let str = "Please locate where 'locate' occurs!";
    let pos = str.indexOf("locate");
    document.getElementById("demo").innerHTML = pos; // 7
  </script>
</body>
</html>
```

```
%%JavaScript%% 0
0 , 1 , 2 ...
```

`lastIndexOf()` 가 ()

```
<!DOCTYPE html>
<html>
<body>
  <h2>JavaScript String Methods</h2>
  <p>The lastIndexOf() method returns the positon of the last occurrence of
a specified text:</p>
  <p id="demo"></p>
  <script>
    let str = "Please locate where 'locate' occurs!";
    let pos = str.lastIndexOf('locate');
    document.getElementById("demo").innerHTML = pos; // 21
  </script>
</body>
</html>
```

가 , indexOf() lastIndexOf() -1 .

```
<!DOCTYPE html>
```

```

<html>
<body>
  <h2>JavaScript String Methods</h2>
  <p>Both indexOf() and lastIndexOf() returns -1 if the text is not
found:</p>
  <p id="demo"></p>
  <script>
    let str = "Please locate where 'locate' occurs!";
    let pos = str.indexOf("John");
    document.getElementById("demo").innerHTML = pos; // -1
  </script>
</body>
</html>

```

```

<!DOCTYPE html>
<html>
<body>
  <h2>JavaScript String Methods</h2>
  <p>The indexOf() method accepts a second parameter as the starting
position for the search:</p>
  <p id="demo"></p>
  <script>
    let str = "Please locate where 'locate' occurs!";
    let pos = str.indexOf("locate", 15); //index[15]
    document.getElementById("demo").innerHTML = pos; // 21
  </script>
</body>
</html>

```

lastIndexOf(() .
, 가 15 , 15 , .

```

<!DOCTYPE html>
<html>
<body>
  <h2>JavaScript String Methods</h2>
  <p>The lastIndexOf() method accepts a second parameter as the starting
position for the search.</p>
  <p>Remember that the lastIndexOf() method searches backwards, so position
15 means start the search at position 15, and search to the beginning.</p>
  <p>Position 15 is position 15 from the beginning</p>
  <p id="demo"></p>
  <script>
    let str = "Please locate where 'locate' occurs!";
    let pos = str.lastIndexOf("locate", 15);
    document.getElementById("demo").innerHTML = pos; // 7
    console.log(str.length); // 36
  </script>

```


- slice(start, end)
- substring(start, end)
- substr(start, length)

The slice() Method

slice() () 가 .
7 12 (13-1) .

```

<!DOCTYPE html>
<html>
<body>
  <h2>JavaScript String Methods</h2>
  <p>The slice() method extract a part of a string and returns the extracted
parts in a new string:</p>
  <p id="demo"></p>
  <script>
    let str = "Apple, Banana, Kiwi";
    let res = str.slice(7, 13); // 7- , 13- , X
    document.getElementById("demo").innerHTML = res; // Banana
  </script>
</body>
</html>

```

: 0 . 0 .

가 , .
-12 -6 .

```

<script>
  let str = "Apple, Banana, Kiwi";
  let res = str.slice(-12, -6);
  document.getElementById("demo").innerHTML = res;
  console.log(str.length);
</script>

```

가 from MDN:

```
str.slice(beginIndex[, endIndex])
```

beginIndex:

-
- , beginIndex = str.length() + beginIndex
-) beginIndex = -3 , strLength + (-3)

endIndex:

- 0
- endIndex

(- > 가 ,
) 가

```
<script>  
let str = "Apple, Banana, Kiwi";  
let res = str.slice(7);  
document.getElementById("demo").innerHTML = res; // Banana, Kiwi  
</script>
```

가 , 가

```
<script>  
let str = "Apple, Banana, Kiwi";  
let res = str.slice(-12);  
document.getElementById("demo").innerHTML = res; // Banana, Kiwi  
</script>
```

Internet Explorer 8

가

The substring() Method

substring() slice()

substring()

```
<script>  
let str = "Apple, Banana, Kiwi";  
let res = str.substring(7, 13); // index7 12  
document.getElementById("demo").innerHTML = res; // Banana  
</script>
```

substring()

The substr() Method

substr() slice()

가

```
<script>
  let str = "Apple, Banana, Kiwi";
  let res = str.substr(7, 10); // index7 10
  document.getElementById("demo").innerHTML = res; // Banana, Ki
</script>
```

, substr()

```
<script>
  let str = "Apple, Banana, Kiwi";
  let res = str.substr(8); // index7
  document.getElementById("demo").innerHTML = res; // anana, Kiwi
</script>
```

가

```
<script>
  let str = "Apple, Banana, Kiwi";
  let res = str.substr(-8); // index-8
  document.getElementById("demo").innerHTML = res; // na, Kiwi
</script>
```

Replacing String Content

replace()

()

```
<!DOCTYPE html>
<html>
<body>
  <h2>JavaScript String Methods</h2>
  <p>Replace "Microsoft" with "W3Schools" in the paragraph below:</p>
  <button onclick="myFunction()">Try it</button>
  <p id="demo">Please visit Microsoft!</p>
  <script>
    function myFunction() {
      let str = document.getElementById("demo").innerHTML;
      let txt = str.replace("Microsoft", "W3Schools");
      document.getElementById("demo").innerHTML = txt;
    }
  </script>
```

```
</script>  
</body>  
</html>
```

```
'replace()'
```

```
, replace()
```

```
<!DOCTYPE html>  
<html>  
<body>  
  <h2>JavaScript String Methods</h2>  
  <p>Replace "Microsoft" with "W3Schools" in the paragraph below:</p>  
  <button onclick="myFunction()">Try it</button>  
  <p id="demo">Please visit Microsoft and Microsoft!</p>  
  <script>  
    function myFunction() {  
      let str = document.getElementById("demo").innerHTML;  
      let txt = str.replace("Microsoft", "W3Schools");  
      document.getElementById("demo").innerHTML = txt;  
    }  
  </script>  
</body>  
</html>
```

```
replace()  
MICROSOFT
```

```
<!DOCTYPE html>  
<html>  
<body>  
  <h2>JavaScript String Methods</h2>  
  <p>Try to replace "Microsoft" with "W3Schools" in the paragraph below:</p>  
  <button onclick="myFunction()">Try it</button>  
  <p id="demo">Please visit Microsoft!</p>  
  <script>  
    function myFunction() {  
      let str = document.getElementById("demo").innerHTML;  
      let txt = str.replace("MICROSOFT", "W3Schools");  
      document.getElementById("demo").innerHTML = txt;  
    }  
  </script>  
  <p><strong>Note:</strong> Nothing will happen. By default, the replace()  
method is case sensitive. Writing MICROSOFT (with upper-case) will not
```



```
work.</P>
</body>
</html>
```

, /i () .

```
<script>
function myFunction() {
  let str = document.getElementById("demo").innerHTML;
  let txt = str.replace(/MICROSOFT/i, "W3Schools");
  document.getElementById("demo").innerHTML = txt;
}
</script>
```

, /g () .

```
<script>
function myFunction() {
  let str = document.getElementById("demo").innerHTML;
  let txt = str.replace(/Microsoft/g, "W3Schools");
  document.getElementById("demo").innerHTML = txt;
}
</script>
```

global match & case-insensitive

```
<script>
function myFunction() {
  let str = document.getElementById("demo").innerHTML;
  let txt = str.replace(/MiCROSOFT/gi, "W3Schools");
  document.getElementById("demo").innerHTML = txt;
}
</script>
```

Converting to Upper and Lower Case

toUpperCase()

```
<!DOCTYPE html>
<html>
<body>
```

```
<p>Convert string to upper case:</p>
<button onclick="myFunction()">Try it</button>
<p id="demo">Bonjour le Monde</p>
<script>
  function myFunction() {
    let text = document.getElementById("demo").innerHTML;
    document.getElementById("demo").innerHTML = text.toUpperCase();
  }
</script>
</body>
</html>
```

toLowerCase()

```
<script>
  function myFunction() {
    let text = document.getElementById("demo").innerHTML;
    document.getElementById("demo").innerHTML = text.toLowerCase();
  }
</script>
```

The concat() Method

concat()

```
<script>
  let text1 = "Impossible is";
  let text2 = "Nothing";
  let text3 = text1.concat(" ", text2); // // text1+ +text2
  document.getElementById("demo").innerHTML = text3; // Impossible is
  Nothing
</script>
```

concat()

```
var text = "Hello" + " " + "World!";
var text = "Hello".concat(" ", "World!");
```

: (immutable) :

String.trim()

trim()

```

<script>
  function myFunction() {
    let str = "   Hello World!   ";
    alert(str.trim());
  }
</script>

```

Internet Explorer 8 trim () 가

IE 8

replace()

```

<script>
  let str = "   Hello World!   ";
  alert(str.replace(/^[^\s\uFEFF\uA0]+|[\s\uFEFF\uA0]+$/g, ''));
</script>

```

replace

JavaScript String.prototype

가

```

<script>
  if (!String.prototype.trim) {
    String.prototype.trim = function () {
      return this.replace(/^[^\s\uFEFF\uA0]+|[\s\uFEFF\uA0]+$/g, '');
    };
  }
  let str = "   Hello World!   ";
  alert(str.trim());
</script>

```

JavaScript String Padding

ECMAScript 2017

가 String

padStart

padEnd

가

MDN 가 :

()

str.padStart(targetLength [, padString])

- targetLength : 가

padString(가)

- "".(U+0020)

```
<script>
  let str = "5";
  str = str.padStart(4, 0);
  document.getElementById("demo").innerHTML = str; // 0005
</script>
```

```
<script>
  let str = "321";
  str = str.padEnd(4, 0);
  document.getElementById("demo").innerHTML = str; // 3210
</script>
```

Internet Explorer

Firefox Safari JavaScript

Extracting String Characters

가 가

- charAt(position)
- charCodeAt(position)
- []

The charAt() Method

charAt() ()

```
<script>
  let str = "HELLO WORLD";
  document.getElementById("demo").innerHTML = str.charAt(6); // W
</script>
```

The charCodeAt() Method

charCodeAt()

UTF-16 (0 65535)

```

<script>
  let str = "HELLO WORLD";
  document.getElementById("demo").innerHTML = str.charCodeAt(0); // 72
</script>

```

Property Access

ECMAScript 5 (2009)

[]


```

* Internet Explorer 7
*
*   가 , %%[]%%   %%undefined%%   , %%charAt()%%
*
*   . %%str[0] = "A"%%   !

```

Converting a String to an Array

split()

(separator) , [0] 

가 "" ,

Complete String Reference

, [Complete JavaScript String Reference](#)

가

, [Javascript](#), [String](#), [Methods](#)

From: <http://rwiki.repia.com/> -

. - 2023.12

Permanent link: http://rwiki.repia.com/doku.php?id=wiki:javascript:javascript_note:js_string_methods&rev=1619167995

Last update: **2022/03/10 19:52**

