

## Obiekt String

Dodał Administrator  
poniedziałek, 15 marzec 2010 03:56

---

Dostępność: od JavaScript 1.1 i JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

Obiekty typu String to nic innego, jak ciągi znaków. Każdy ciąg ujęty w znaki cudzysłowu lub apostrofu, np.:

"abc";

'def'

jest traktowany jako obiekt typu String, co oznacza, że w stosunku do niego mogą być wywoływane metody, np.:

```
var c = "abc".charAt(1);
```

Począwszy od wersji 1.1 JavaScript i 2.0 JScript, obiekt typu String może zostać również utworzony przez wywołanie konstruktora, np.:

```
var str = new String("abc");
```

W ciągu znaków można umieszczać znaki specjalne, przedstawione w tabeli 3.3.

Tabela 3.3. Sekwencje znaków specjalnych

Sekwencja

Znaczenie

\b

backspace (ang. backspace)

\n

nowa linia (ang. new line)

\r

powrót karetki (ang. carriage return)

\f

nowa strona (ang. form feed)

\t

tabulacja (ang. horizontal tab)

## Obiekt String

Dodał Administrator  
poniedziałek, 15 marzec 2010 03:56

---

\&quot;

cudzysłów (ang. double quote)

\'

apostrof (ang. single quote)

\\

lewy ukośnik (ang. backslash)

Jeśli zatem w napisie ma się znaleźć np. cudzysłów, to należy go poprzedzić znakiem \, podobnie będzie z apostrofem, np.:

```
document.write("&quot;To jest cudzysłów &quot;&quot;);
```

```
document.write("&quot;To jest apostrof '&quot;);
```

Do łączenia (konkatenacji) łańcuchów znakowych służy operator +. Używa się go podobnie jak operatora arytmetycznego +, np.:

```
var str = "&quot;abc&quot; + &quot;def&quot;;
```

```
var str1 = "&quot;abc&quot;; str2 = &quot;def&quot;;
```

```
var str3 = str1 + str2;
```

Do łączenia może być także wykorzystywany operator łączony +=, np.:

```
var str = "&quot;abc&quot;;
```

```
str1 += "&quot;def&quot;;
```

### **Właściwości obiektu String**

#### **Właściwość length**

Składnia: string.length

Dostępność: od JavaScript 1.0, JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

Właściwość length pozwala określić długość ciągu znaków, np. po wykonaniu instrukcji:

```
var tekst = "&quot;abc&quot;;
```

```
var count = tekst.length;
```

zmienna count będzie zawierała wartość 3. Ponieważ ciąg znaków w trakcie przetwarzania przez aparat wykonawczy JavaScript jest zamieniany na obiekt, odwołanie do jego właściwości (oraz metod) może być przeprowadzone bezpośrednio, np.:

## Obiekt String

Dodał Administrator  
poniedziałek, 15 marzec 2010 03:56

---

var count = &quot;abc&quot;.length; **Metody formatujące ciągi**

Metody formatujące ciągi powodują dodanie do danego ciągu znaczników HTML wpływających na wygląd bądź funkcję tekstu (np. pogrubienie lub utworzenie odnośnika). **Metoda anchor**

Składnia: string.anchor(param)

Dostępność: od JavaScript 1.0 i JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

Metoda anchor zwraca ciąg znaków, w którym ciąg reprezentowany przez string został ujęty w znacznik <A> z parametrem NAME równym param. Przykładowe wywołanie:

```
var str = &quot;abc&quot;.anchor(&quot;def&quot;);
```

spowoduje umieszczenie w zmiennej str znacznika <a> w postaci:

```
<A NAME=&quot;def&quot;>abc</A>
```

dla przeglądarek IE, OP i

```
<a name=&quot;def&quot;>abc</a>
```

dla przeglądarek FF, NN. **Metoda big**

Składnia: string.big()

Dostępność: od JavaScript 1.0 i JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

Metoda big zwraca ciąg znaków, w którym ciąg reprezentowany przez string został ujęty w znacznik <BIG>. Przykładowe wywołanie:

```
var str = &quot;abc&quot;.big();
```

spowoduje umieszczenie w zmiennej str znacznika <BIG> w postaci:

```
<BIG>abc</BIG>
```

dla przeglądarek IE, OP i

```
<big>abc</big>
```

dla przeglądarek FF, NN. **Metoda blink**

Składnia: string.blink()

Dostępność: od JavaScript 1.0 i JScript 1.0

## Obiekt String

Dodał Administrator  
poniedziałek, 15 marzec 2010 03:56

---

Przeglądarki: IE3, FF, NN2, OP7

Metoda `blink` zwraca ciąg znaków, w którym ciąg reprezentowany przez `string` został ujęty w znacznik `<BLINK>`.  
Przykładowe wywołanie:

```
var str = "abc".blink();
```

spowoduje umieszczenie w zmiennej `str` znacznika `<BLINK>` w postaci:

```
<BLINK>abc</BLINK>
```

dla przeglądarek IE, OP i

```
<blink>abc</blink>
```

dla przeglądarek FF, NN. **Metoda `bold`**

Składnia: `string.bold()`

Dostępność: od JavaScript 1.0 i JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

Metoda `bold` zwraca ciąg znaków, w którym ciąg reprezentowany przez `string` został ujęty w znacznik `<B>`.  
Przykładowe wywołanie:

```
var str = "abc".bold();
```

spowoduje umieszczenie w zmiennej `str` znacznika `<B>` w postaci:

```
<B>abc</B>
```

dla przeglądarek IE, OP i

```
<b>abc</b>
```

dla przeglądarek FF, NN. **Metoda `fixed`**

Składnia: `string.fixed()`

Dostępność: od JavaScript 1.0 i JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

Metoda `fixed` zwraca ciąg znaków, w którym ciąg reprezentowany przez `string` został ujęty w znacznik `<TT>`.  
Przykładowe wywołanie:

```
var str = "abc".fixed();
```

spowoduje umieszczenie w zmiennej `str` znacznika `<TT>` w postaci:

```
<TT>abc</TT>
```

## Obiekt String

Dodał Administrator  
poniedziałek, 15 marzec 2010 03:56

---

dla przeglądarek IE, OP i

```
<tt>abc</tt>
```

dla przeglądarek FF, NN. **Metoda fontcolor**

Składnia: string.fontcolor(kolor)

Dostępność: od JavaScript 1.0 i JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

Metoda fontcolor zwraca ciąg znaków, w którym ciąg reprezentowany przez string został ujęty w znacznik <FONT>, z parametrem COLOR równym argumentowi kolor. Przykładowe wywołanie:

```
var str = &quot;abc&quot;.fontcolor(&quot;red&quot;);
```

spowoduje umieszczenie w zmiennej str znacznika <A> w postaci:

```
<FONT COLOR=&quot;red&quot;>abc</FONT>
```

dla przeglądarek IE, OP i

```
<font color=&quot;red&quot;>abc</font>
```

dla przeglądarek FF, NN. **Metoda fontsize**

Składnia: string.fontSize(rozmiar)

Dostępność: od JavaScript 1.0 i JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

Metoda fontsize zwraca ciąg znaków, w którym ciąg reprezentowany przez string został ujęty w znacznik <FONT>, z parametrem SIZE równym argumentowi rozmiar. Przykładowe wywołanie:

```
var str = &quot;abc&quot;.fontSize(10);
```

spowoduje umieszczenie w zmiennej str znacznika <FONT> w postaci:

```
<FONT SIZE=&quot;10&quot;>abc</FONT>
```

dla przeglądarek IE, OP i

```
<font size=&quot;10&quot;>abc</font>
```

dla przeglądarek FF, NN. **Metoda italics**

Składnia: string.italics()

Dostępność: od JavaScript 1.0 i JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

## Obiekt String

Dodał Administrator  
poniedziałek, 15 marzec 2010 03:56

---

Metoda `italics` zwraca ciąg znaków, w którym ciąg reprezentowany przez string został ujęty w znacznik `<I>`.  
Przykładowe wywołanie:

```
var str = "abc".italics();
```

spowoduje umieszczenie w zmiennej `str` znacznika `<I>` w postaci:

```
<I>abc</I>
```

dla przeglądarek IE, OP i

```
<I>abc</I>
```

dla przeglądarek FF, NN. **Metoda link**

Składnia: `string.link(param)`

Dostępność: od JavaScript 1.0 i JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

Metoda `link` zwraca ciąg znaków, w którym ciąg reprezentowany przez string został ujęty w znacznik `<A>` z parametrem `HREF` równym `param`. Przykładowe wywołanie:

```
var str = "abc".link("http://nazwa.domeny");
```

spowoduje umieszczenie w zmiennej `str` znacznika `<A>` w postaci:

```
<A HREF="http://nazwa.domeny">abc</A>
```

dla przeglądarek IE, OP i

```
<a href="http://nazwa.domeny">abc</a>
```

dla przeglądarek FF, NN. **Metoda small**

Składnia: `string.small()`

Dostępność: od JavaScript 1.0 i JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

Metoda `small` zwraca ciąg znaków, w którym ciąg reprezentowany przez string został ujęty w znacznik `<SMALL>`.  
Przykładowe wywołanie:

```
var str = "abc".small();
```

spowoduje umieszczenie w zmiennej `str` znacznika `<SMALL>` w postaci:

```
<SMALL>abc</SMALL>
```

dla przeglądarek IE, OP i

## Obiekt String

Dodał Administrator  
poniedziałek, 15 marzec 2010 03:56

---

`<small>abc</small>`

dla przeglądarek FF, NN. **Metoda strike**

Składnia: `string.strike()`

Dostępność: od JavaScript 1.0 i JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

Metoda `strike` zwraca ciąg znaków, w którym ciąg reprezentowany przez `string` został ujęty w znacznik `<STRIKE>`. Przykładowe wywołanie:

```
var str = "abc".strike();
```

spowoduje umieszczenie w zmiennej `str` znacznika `<STRIKE>` w postaci:

```
<STRIKE>abc</STRIKE>
```

dla przeglądarek IE, OP i

```
<strike>abc</strike>
```

dla przeglądarek FF, NN. **Metoda sub**

Składnia: `string.sub()`

Dostępność: od JavaScript 1.0 i JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

Metoda `sub` zwraca ciąg znaków, w którym ciąg reprezentowany przez `string` został ujęty w znacznik `<SUB>`. Przykładowe wywołanie:

```
var str = "abc".sub();
```

spowoduje umieszczenie w zmiennej `str` znacznika `<SUB>` w postaci:

```
<SUB>abc</SUB>
```

dla przeglądarek IE, OP i

```
<sub>abc</sub>
```

dla przeglądarek FF, NN. **Metoda sup**

Składnia: `string.sup()`

Dostępność: od JavaScript 1.0 i JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

Metoda `sup` zwraca ciąg znaków, w którym ciąg reprezentowany przez `string` został ujęty w znacznik `<SUP>`.

## Obiekt String

Dodał Administrator  
poniedziałek, 15 marzec 2010 03:56

---

Przykładowe wywołanie:

```
var str = "abc".sup();
```

spowoduje umieszczenie w zmiennej str znacznika <SUP> w postaci:

```
<SUP>abc</SUP>
```

dla przeglądarek IE, OP i

```
<sup>abc</sup>
```

dla przeglądarek FF, NN. **Metody przetwarzające ciągi**

### Metoda charAt

Składnia: string.charAt(indeks)

Dostępność: od JavaScript 1.0 i JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

Metoda charAt zwraca znak ciągu znajdujący się pod indeksem wskazanym przez argument indeks, np. po wykonaniu instrukcji:

```
var c = "abc".charAt(1);
```

w zmiennej c zostanie zapisany znak b (indeksowanie ciągu rozpoczyna się od 0, czyli pierwszy znak ma indeks 0). Należy zwrócić uwagę, że sekwencje znaków specjalnych tworzą w rzeczywistości jeden znak, czyli w wyniku wykonania instrukcji:

```
var c1 = "\n".charAt(0);
```

```
var c2 = "\n".charAt(1);
```

otrzymamy przypisanie zmiennej c1 znaku '\', a zmiennej c2 — znaku nowego wiersza. **Metoda charCodeAt(indeks)**

Składnia: string.charCodeAt(indeks)

Dostępność: od JavaScript 1.2 i JScript 5.5

Przeglądarki: IE5.5, FF, NN4, OP7

Metoda charCodeAt zwraca kod znaku znajdującego się pod indeksem indeks. W wersji JavaScript 1.2 jest to kod ISO-Latin1, a od wersji 1.3 (oraz w JScript) — kod Unicode. Przykładowo wykonanie instrukcji:

```
var code = "abc".charCodeAt(1);
```

spowoduje przypisanie zmiennej code wartości 98 (kod litery b).

W przypadku wystąpienia w ciągu sekwencji znaków specjalnych obowiązują te same zasady, co dla metody charAt, czyli wywołanie:

```
var c = "\n".charCodeAt(0);
```



## Obiekt String

Dodał Administrator  
poniedziałek, 15 marzec 2010 03:56

---

spowoduje przypisanie zmiennej c wartości 39 (kod znaku '). **Metoda concat**

Składnia: `string.concat(string2, string3, ... , stringn)`

Dostępność: od JavaScript 1.2 i JScript 3.0

Przeglądarki: IE4, FF, NN4, OP7

Metoda `concat` pozwala na łączenie łańcuchów znakowych. Wynikiem działania metody jest ciąg powstały z połączenia łańcucha `string` i wszystkich łańcuchów przekazanych w postaci argumentów, np. po wykonaniu instrukcji:

```
var str1 = "abc";, str2 = "def";;
```

```
var str3 = "123".concat(str1, str2);
```

zmienna `str3` będzie zawierała ciąg `123abcdef`. **Metoda fromCharCode**

Składnia: `String.fromCharCode(kod1, kod2, ... , kodn)`

Dostępność: od JavaScript 1.2 i JScript 3.0

Przeglądarki: IE4, FF, NN4, OP7

Metoda `fromCharCode` zwraca ciąg znaków, których kody zostały przekazane w postaci argumentów. Jest to metoda statyczna, do jej wywołania nie jest zatem konieczne istnienie obiektu typu `String`. Przykładowo: gdyby istniała potrzeba uzyskania ciągu znaków `ABC`, czyli znaków o kodach 65, 66 i 67, można by wykorzystać instrukcję:

```
var str = String.fromCharCode(65, 66, 67); Metoda indexOf
```

Składnia: `string.indexOf(value [, index])`

Dostępność: od JavaScript 1.0 i JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

Metoda `indexOf` zwraca indeks wystąpienia ciągu `value` w ciągu `string`. Jeżeli podany zostanie opcjonalny drugi argument, przeszukiwanie ciągu `string` rozpocznie się od znaku znajdującego się pod indeksem `index`. Jeżeli ciąg `value` nie zostanie znaleziony, metoda zwraca wartość `-1`. Przykładowo wywołanie:

```
var indeks = "abcabcabc".indexOf("bc");
```

spowoduje przypisanie zmiennej `indeks` wartości 1, a wywołanie:

```
var indeks = "abcabcabc".indexOf("bc", 5);
```

przypisanie zmiennej `indeks` wartości 7. **Metoda lastIndexOf**

Składnia: `string.lastIndexOf(value [, index])`

Dostępność: od JavaScript 1.0 i JScript 1.0

## Obiekt String

Dodał Administrator  
poniedziałek, 15 marzec 2010 03:56

---

Przeglądarki: IE3, FF, NN2, OP7

Metoda `lastIndexOf` działa podobnie jak `indexOf`, z tą różnicą, że ciąg string przeszukiwany jest od końca. Na przykład wywołanie:

```
var indeks = "abcabcabc".lastIndexOf("bc");
```

da w rezultacie wynik 7, a wywołanie:

```
var indeks = "abcabcabc".lastIndexOf("bc", 5);
```

da w rezultacie wynik 4. **Metoda `localeCompare`**

Składnia: `string.localeCompare(str)`

Dostępność: od JavaScript 1.5 i JScript 5.5

Przeglądarki: IE5.5, FF, NN4, OP7

Metoda porównuje ciąg string i ciąg przekazany jako argument `str`. Przy porównywaniu uwzględniane są ustawienia narodowe i językowe przeglądarki. Zwracane są następujące wartości:

-

wartość mniejsza od zera (typowo `-1`) — gdy ciąg string jest mniejszy od `str` (tzn. w porządku alfabetycznym powinien znaleźć się przed ciągiem `str`);

-

0 — gdy ciągi są równe (takie same);

-

wartość większa od zera (typowo `1`) — gdy ciąg string jest większy od `str` (tzn. w porządku alfabetycznym powinien znaleźć się za ciągiem `str`).

Przykładowo wywołanie:

```
var i = "abc".localeCompare("abc");
```

spowoduje przypisanie zmiennej `i` wartości 0. Wywołanie:

```
var i = "bcd".localeCompare("abc");
```

spowoduje przypisanie zmiennej `i` wartości 1. Wywołanie:

```
var i = "abc".localeCompare("bcd");
```

spowoduje przypisanie zmiennej `i` wartości `-1`. **Metoda `match`**

Składnia: `string.match(wyrReg)`

Dostępność: od JavaScript 1.2 i JScript 3.0

Przeglądarki: IE4, FF, NN4, OP7

## Obiekt String

Dodał Administrator

poniedziałek, 15 marzec 2010 03:56

---

Metoda `match` zwraca tablicę zawierającą wszystkie podciągi ciągu string zgodne z wyrażeniem regularnym przekazany w postaci argumentu `wyrReg`. Jeżeli nie istnieje żaden taki podciąg, zwracana jest wartość `null`. Metoda pozwala również na pracę ze zwykłymi ciągami, a zatem możemy stwierdzić, czy w danym ciągu występuje pewien podciąg. Jeśli tak jest, metoda zwróci jednoelementową tablicę, która w komórce o indeksie 0 będzie zawierała dany podciąg, jeśli nie — zwróci wartość `null`. A więc po wykonaniu instrukcji:

```
var str = "javascript".match("ava");
```

zmienna `str` będzie zawierała jednoelementową tablicę z ciągiem `ava` (ponieważ ciąg `ava` jest podciągiem ciągu `javascript`), a po wykonaniu instrukcji:

```
var str = "javascript".match("abc");
```

zmienna `str` będzie zawierała wartość `null` (ponieważ ciąg `abc` nie jest podciągiem ciągu `javascript`). **Metoda `replace`**

Składnia: `string.replace(wyrReg, tekst)`

Dostępność: od JavaScript 1.2 i JScript 3.0

Przeglądarki: IE3(5.5), FF, NN4, OP7

Metoda `replace` zwraca ciąg, w którym podciągi opisane przez wyrażenie regularne `wyrReg` zostały zamienione na ciąg znaków reprezentowany przez `tekst`. Na przykład po wykonaniu instrukcji:

```
var str = "Ala ma kota. Ala ma też psa.;"
```

```
var str2 = str.replace(/Ala/gi, "Ola");
```

występujące w ciągu imię `Ala` zostanie zamienione na `Ola`.

Ciąg `tekst` może zawierać sekwencje znaków specjalnych zebrane w tabeli 3.4 (dostępne w JavaScript od wersji 1.2 i JScript od wersji 5.5).

Tabela 3.4. Sekwencje specjalne dla funkcji `replace`

Sekwencja

Znaczenie

`$$`

znak `$`

`$&`

podciąg pasujący do wyrażenia `wyrReg`

`$``

podciąg znajdujący się przed ciągiem `$&`

## Obiekt String

Dodał Administrator  
poniedziałek, 15 marzec 2010 03:56

---

\$'

podciąg znajdujący się za ciągiem \$&

\$n lub \$nn

n-ty podciąg wyrażenia wyrReg (patrz przykład poniżej)

Przykład wykorzystania sekwencji specjalnej \$n:

```
var str = "Ala ma kota. Ala ma też psa.;"
```

```
var str2 = str.replace(/(A)(l)(a)/gi, "$3$2$1");
```

**Metoda search**

Składnia: `string.search(wyrReg)`

Dostępność: od JavaScript 1.2 i JScript 3.0

Przeglądarki: IE4, FF, NN4, OP7

Metoda `search` sprawdza, czy ciąg opisany przez wyrażenie regularne `wyrReg` występuje w ciągu `string`. Jeśli tak, zwracany jest indeks wystąpienia, jeśli nie, zwracana jest wartość `-1` (działa więc podobnie do `indexOf`).  
**Metoda slice**

Składnia: `string.slice(start[, end])`

Dostępność: od JavaScript 1.0 i JScript 3.0

Przeglądarki: IE4, FF, NN2, OP7

Metoda `slice` zwraca podciąg ciągu `string` rozpoczynający się od indeksu wskazywanego przez `start` i kończący się wraz z końcem ciągu `string` lub też, jeśli został podany parametr `end`, ze znakiem wskazanym przez argument `end`. Przykładowe wywołania:

```
var str1 = "abcdefg".slice(3);
```

```
var str2 = "abcdefg".slice(3, 5);
```

spowodują przypisanie zmiennej `str1` wartości `defg`, a zmiennej `str2` wartości `de`.

Można przy tym stosować argumenty o wartościach ujemnych, które są interpretowane jako przesunięcia względem końca ciągu głównego. Przykładowo wywołanie:

```
var str = "abcdefg".slice(2, -3);
```

spowoduje przypisanie zmiennej `str` ciągu `cd`. Jest to ciąg rozpoczynający się w znaku o indeksie 2, licząc od początku, i kończący się w znaku o indeksie 3, licząc od końca. Identyczny efekt da wywołanie:

```
var str = "abcdefg".slice(-5, -3);
```

## Obiekt String

Dodał Administrator  
poniedziałek, 15 marzec 2010 03:56

---

Będzie to bowiem ciąg rozpoczynający się w znaku o indeksie 5, licząc od końca, i kończący się w znaku o indeksie 3, również licząc od końca. **Metoda split**

Składnia: `string.split([separator[, limit]])`

Dostępność: od JavaScript 1.1 i JScript 3.0

Przeglądarki: IE4, FF, NN3, OP7

Metoda `split` dzieli ciąg znaków `string` na podciągi względem parametru `separator` i zwraca je w postaci tablicy. Parametr `separator` może być ciągiem znaków lub wyrażeniem regularnym, jeżeli nie zostanie podany, zwrócony zostanie pełny ciąg `string`. Parametr `limit` jest wartością całkowitą wskazującą maksymalną liczbę elementów tablicy wynikowej. Jeżeli zatem istnieje potrzeba np. podzielenia ciągu względem znaków spacji (czyli wydzielenia odrębnych słów), można zastosować konstrukcję widoczną na listingu 3.15.

Listing 3.15. Ilustracja działania metody `split`

```
<script type="text/javascript">
var tab = "ab cd ef gh".split(" ");
for(indeks in tab){
    document.write(tab[indeks] + "<br />");
}
</script>
```

Jeżeli jako separator zostanie użyty pusty ciąg znaków `" "`, ciąg źródłowy zostanie podzielony na poszczególne znaki. Np. wykonanie instrukcji:

```
var tab = "dwa".split("");
```

spowoduje powstanie tablicy `tab` zawierającej trzy komórki z ciągami `d`, `w` i `a`. **Metoda substr**

Składnia: `string.substr(indeks, ile)`

Dostępność: od JavaScript 1.0 i JScript 3.0

Przeglądarki: IE4, FF, NN2, OP7

Metoda `substr` zwraca podciąg ciągu `string` rozpoczynający się od znaku o indeksie `indeks`, zawierający liczbę znaków wskazywanych przez parametr `ile`. Jeżeli parametr `ile` nie zostanie podany, zwracane są wszystkie znaki od indeksu `indeks` do końca ciągu `string`, np. wywołanie:

```
"abcdefg".substr(2, 3);
```

da w wyniku ciąg `cde`, a wywołanie:

```
"abcdefg".substr(4);
```

da w wyniku ciąg `efg` (wszystkie znaki, począwszy od tego o indeksie 4, aż do końca ciągu głównego). **Metoda substring**

## Obiekt String

Dodał Administrator  
poniedziałek, 15 marzec 2010 03:56

---

Składnia: `string.substring(indeks1[, indeks2])`

Dostępność: od JavaScript 1.0 i JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

Metoda `substring` zwraca podciąg ciągu `string` rozpoczynający się od znaku o indeksie `indeks1` i kończący się przed znakiem o indeksie `indeks2`, np. wywołanie:

```
var tab = "abcdefg".substring(2, 5);
```

da w wyniku ciąg `cde`. Jeżeli nie zostanie podany drugi argument, pobrane zostaną wszystkie znaki, począwszy od indeksu wskazywanego przez pierwszy argument, aż do końca ciągu, np. wywołanie:

```
var tab = "abcdefg".substring(4);
```

da w wyniku ciąg `efg`.

Zachowanie tej metody jest podobne do `slice`. Różnica jest taka, że jeżeli pierwszy argument jest większy od drugiego (czyli indeks początkowy jest większy od końcowego), ich kolejność zostanie zamieniona. Zatem wywołanie `substring(4, 2)` zostanie potraktowane jak `substring(2, 4)`. **Metoda `toLocaleLowerCase`**

Składnia: `string.toLocaleLowerCase()`

Dostępność: od JavaScript 1.5 i JScript 5.5

Przeglądarki: IE5.5, FF, NN6, OP7

Metoda `toLowerCase` zwraca ciąg, w którym wszystkie znaki ciągu `string` zostały zamienione na małe litery z uwzględnieniem ustawień narodowych i językowych. W większości przypadków jest to równoznaczne z wywołaniem metody `toLowerCase`. Wywołanie metody nie wpływa na zawartość ciągu `string`. **Metoda `toLocaleUpperCase`**

Składnia: `string.toLocaleLowerCase()`

Dostępność: od JavaScript 1.5 i JScript 5.5

Przeglądarki: IE5.5, FF, NN6, OP7

Metoda `toLowerCase` zwraca ciąg, w którym wszystkie znaki ciągu `string` zostały zamienione na wielkie litery z uwzględnieniem ustawień narodowych i językowych. W większości przypadków jest to równoznaczne z wywołaniem metody `toUpperCase`. Wywołanie metody nie wpływa na zawartość ciągu `string`. **Metoda `toLowerCase`**

Składnia: `string.toLowerCase()`

Dostępność: od JavaScript 1.0 i JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

Metoda `toLowerCase` zwraca ciąg, w którym wszystkie znaki ciągu `string` zostały zamienione na małe litery. Wywołanie metody nie wpływa na zawartość ciągu `string`. **Metoda `toUpperCase`**

## Obiekt String

Dodał Administrator  
poniedziałek, 15 marzec 2010 03:56

---

Składnia: `string.toUpperCase()`

Dostępność: od JavaScript 1.0 i JScript 1.0

Przeglądarki: IE3, FF, NN2, OP7

Metoda `toUpperCase` zwraca ciąg, w którym wszystkie znaki ciągu `string` zostały zamienione na wielkie litery. Wywołanie metody nie wpływa na zawartość ciągu `string`. **Metoda `toString`**

Składnia: `string.toString()`

Dostępność: od JavaScript 1.1 i JScript 2.0

Przeglądarki: IE3, FF, NN2, OP7

Metoda `toString` zwraca ciąg znaków będący reprezentacją obiektu, na rzecz którego została wywołana. W przypadku obiektu klasy `String` będzie to reprezentowany przez ten obiekt ciąg znaków. **Metoda `valueOf`**

Składnia: `string.valueOf()`

Dostępność: od JavaScript 1.1 i JScript 2.0

Przeglądarki: IE4, FF, NN4, OP7

Metoda `valueOf` zwraca podstawową wartość przechowywaną przez obiekt. W przypadku obiektu typu `String` działanie tej metody jest równoznaczne z wywołaniem `string.toString()` i zwraca zawarty w obiekcie ciąg znaków.