

Obiekty wbudowane w JavaScript

String

Właściwości

length

```
tekst = "Obiekty języka JavaScript"
dl = tekst.length
```

Metody

metoda substring()

zwraca podzbiór tego łańcucha. Jej parametry określają położenie początku i końca podzbioru.

```
tekst = "Obiekty języka JavaScript"
x = tekst.substring (15,19)
```

x będzie mieć **wartość 26**

metoda toUpperCase()

zamienia wszystkie litery ciągu na duże

```
tekst = "Obiekty języka JavaScript"
x = tekst.toUpperCase()
```

Wyświetlając w konsoli zobaczymy taki oto wynik:

```
OBIEKTY JÄ™ZYKA JAVASCRIPT
```

metoda toLowerCase()

zamienia wszystkie litery ciągu na małe

```
tekst = "Obiekty języka JavaScript"
x = tekst.toLowerCase()
```

Wyświetlając w konsoli zobaczymy taki oto wynik:

```
obiekty jÄ™zyka javascript
```

Date

Obiekt ten służy do przechowywania wartości daty i czasu. Przy jego pomocy można odczytać wartość daty i czasu, można też rozłożyć te wartości na części, odczytując oddzielnie dzień, miesiąc, rok itp. Można również te części niezależnie modyfikować.

Odczytanie bieżącej daty

Aby odczytać bieżącą datę i czas, należy utworzyć obiekt Date bez parametrów.

```
var dat_cz = new Date();
```

Można utworzyć obiekt z określoną liczbą parametrów. Tych parametrów może być od dwóch do siedmiu (rok, miesiąc, dzień, godzina, minuty, sekundy, milisekundy). Taki obiekt

będzie zawierał ściśle określoną wartość daty i godziny.

```
var data = new Date(2013,2,27);
```

W języku JavaScript wartości daty i czasu są przechowywane w formacie timestamp, czyli jako liczba milisekund, które upłynęły od północy 1 stycznia 1970 roku.

Do konwersji obiektu Date na tekst służy kilka funkcji:

- **toString()** - zwraca datę, czas oraz informacje o strefie czasowej w języku angielskim,
- **toLocaleString()** - zwraca datę i czas dla bieżących ustawień regionalnych,
- **toUTCString()** - zwraca datę, czas oraz informacje o strefie czasowej dla formatu UTC (Universal Coordinated Time),
- **toGMTString()** - działa jak funkcja toUTCString(),
- **toDateString()** - zwraca tylko datę w języku angielskim,
- **toLocaleDateString()** - zwraca tylko datę dla bieżących ustawień regionalnych,
- **toTimeString()** - zwraca tylko czas w języku angielskim,
- **toLocaleTimeString()** - zwraca tylko czas dla bieżących ustawień regionalnych.

Uwaga!

Dla funkcji toString() i toLocaleString() różne przeglądarki zwracają wyniki w różny sposób.

Przykład

Jednym z najczęściej spotykanych przykładów wykorzystania JavaScript na stronie WWW jest wyświetlanie daty i czasu jako elementu strony internetowej.

```
<html>
<head>
<title>JavaScript - Data i czas</title>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<body>
<h2>Wyświetlam bieżącą datę i czas</h2>
<p>
<script type="text/javascript">
data_n = new Date();
data_l = data_n.toString();
data_u = data_n.toGMTString();
data_r = data_n.toLocaleString();
document.write("<b>Czas lokalny:</b> " + data_l + "<br>");
document.write("<b>Czas uniwersalny:</b> " + data_u + "<br>");
document.write("<b>Czas regionalny:</b> " + data_r + "<br>");

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

Przeglądarka Chrome wyświetli w taki oto sposób powyższą stronę

Wyświetlam bieżącą datę i czas

Czas lokalny: Mon Nov 28 2022 16:29:22 GMT+0100 (czas środkowoeuropejski standardowy)

Czas uniwersalny: Mon, 28 Nov 2022 15:29:22 GMT

Czas regionalny: 28.11.2022, 16:29:22

Array

Tabele służą do przechowywania wielu zmiennych. W języku JavaScript do pracy z tabelami można używać wbudowanego **obiektu Array**. Posiada on metody do manipulowania tablicami zmiennych.

Tworzenie nowej tablicy

Aby utworzyć nową tablicę, należy zadeklarować obiekt Array w postaci:

```
var NazwaTablicy = new Array()
```

lub

```
var NazwaTablicy = []
```

Jeżeli w nawiasach zostanie podana liczba **n**, to zostanie utworzona tablica zawierająca **n** pustych elementów

```
var Tab1 = new Array(10)
var Tab2 = [15]
```

Tablicę można również tworzyć, wstawiając do niej konkretne wartości.

```
var Tab3 = new Array('Anna', 'Adam', 'Piotr', 'Ewa')
var Tab4 = ['Paweł', 'Marcin', 'Ela']
```

Żeby uzyskać dostęp do elementów tablicy, należy podać numer indeksu danego elementu. Elementy są indeksowane od zera.

```
document.write(Tab3[2])
```

Aby dodać nową wartość do tablicy, należy przypisać tę wartość do odpowiedniego indeksu tablicy

```
var Tab5 = new Array('kot', 'pies', 'koń')
Tab5[3] = 'mysz';
Tab5[4] = 'chomik';
document.write(Tab5[0] + ' i ' + Tab5[3])
```

Ilość elementów tablicy

Dzięki właściwości **length** można określić, z ilu elementów składa się tablica. Jest to bardzo przydatna właściwość, szczególnie gdy chcemy utworzyć pętlę odczytującą wszystkie elementy tablicy.

```
var Tablica_N = new Array('Anna', 'Adam', 'Piotr', 'Ewa', 'Paweł', 'Marcin',  
'Ela')  
for (i = 0; i<tablica_N.length; i++)  
{  
document.write(Tablica_N[i] + "<br>");  
}
```

W przeglądarce zobaczymy

Anna
Adam
Piotr
Ewa
Paweł
Marcin
Ela

Łączenie elementów tablicy

Za pomocą metody **join()** można łączyć elementy tablicy w jeden tekst. W metodzie tej można opcjonalnie podać parametr, który określi znak oddzielający kolejne elementy tablicy. Jeżeli nie zostanie podana wartość tego parametru, domyślnym znakiem będzie przecinek.

```
var Tablica = new Array('Anna', 'Adam', 'Piotr');  
document.write(Tablica.join() + "<br>");  
document.write(Tablica.join(" - ") + "<br>");
```

W przeglądarce zobaczymy:

Anna,Adam,Piotr
Anna - Adam - Piotr

Odwracanie kolejności elementów tablicy

Za pomocą metody **reverse()** można odwrócić kolejność elementów tablicy.

```
var Tablica = new Array('Anna', 'Adam', 'Piotr');  
document.write(Tablica.join() + "<br>");  
Tablica.reverse()  
document.write(Tablica.join() + "<br>");
```

Sortowanie

Do sortowania elementów tablicy służy metoda **sort()**.

```
var Tablica = new Array('Paweł', 'Anna', 'Maria', 'Adam', 'Piotr');
Tablica.sort()
document.write(Tablica.join());
```

Domyślnie tablica jest sortowana **leksykograficznie**. Powoduje to, że liczba 12459 będzie mniejsza od 4567, ponieważ cyfra na pierwszej pozycji jest mniejsza.

Aby temu zaradzić, można sortować tablicę według własnych kryteriów. Należy skorzystać z dodatkowego parametru metody sort(). Parametrem będzie własna funkcja sortująca. Tworząc taką funkcję, należy pamiętać o trzech zasadach, które muszą być spełnione:

- jeżeli funkcja(a, b) zwróci wartość mniejszą od 0, to wartości a zostanie nadany indeks mniejszy od indeksu przyznanego wartości b,
- jeżeli funkcja(a, b) zwróci wartość równą 0, to wartości indeksów pozostaną bez zmian,
- jeżeli funkcja(a, b) zwróci wartość większą od 0, to wartości a zostanie nadany indeks większy od indeksu przyznanego wartości b.

Stosując się do tych zasad, można tworzyć własne metody sortowania w tablicy.

W podanym przykładzie została zdefiniowana funkcja porownaj(a, b) , która zwróci wartość mniejszą od zera, równą zero lub większą od zera. W zależności od zwróconej wartości elementy tablicy zostaną uporządkowane od wartości najmniejszej do największej.

```
function porownaj(a,b)
{
    return a - b
}

var Tablica = new Array(27, 100, 10, 450, 1654, 320);
document.write('Bez sortowania: ' + Tablica.join() );

document.write('Sortowanie domyślne: ');
Tablica.sort()
document.write(Tablica.join());

document.write('Sortowanie poprawne: ');
Tablica.sort(porownaj)
document.write(Tablica.join() );
```

W przeglądarce zobaczymy:

Bez sortowania: 27,100,10,450,1654,320

Sortowanie domyślne: 10,100,1654,27,320,450

Sortowanie właściwe: 10,27,100,320,450,1654

Zadania

1. Utwórz skrypt pobierający dowolny tekst, a następnie wyświetlający długość tego tekstu.
2. Utwórz skrypt pobierający dowolny tekst pisany dużymi i małymi literami, a następnie wyświetli ten tekst dużymi literami.
3. Utwórz skrypt pobierający dowolny tekst pisany dużymi i małymi literami, a następnie wyświetli ten tekst małymi literami.
4. Utwórz skrypt który z łańcucha o twoim imieniu i nazwisku, wytnie i wyświetli tylko twoje imię.
5. Utwórz skrypt wyświetlający aktualną datę i godzinę.
6. Utwórz skrypt wyświetlający tylko aktualną datę.
7. Utwórz skrypt wyświetlający tylko aktualną godzinę.
8. Utwórz skrypt zawierający tablicę przechowującą imiona osób z grupy.
9. Utwórz skrypt wyświetlający za pomocą pętli wszystkie imiona osób z zadania 8.
10. Utwórz skrypt wyświetlający bez użycia pętli wszystkie imiona osób z zadania 8.
11. Utwórz skrypt wyświetlający posortowane imiona osób z zadania 8.