



JS

Wprowadzenie do JavaScript

Czym jest JS

JS

JavaScript to współczesny język programowania, który jest niemal rówieśnikiem języków **HTML** i **CSS**. Mówiąc nieco nieprecyzyjnie, zasadniczo pozwala on na dodawanie do dokumentów interaktywności. Krótka lista rzeczy, które można zrobić za pomocą JavaScriptu, zawiera następujące elementy:

- nasłuchiwanie zdarzeń, takich jak kliknięcie myszą, i wykonanie jakiejś czynności;
- modyfikowanie kodu HTML i CSS strony po jej załadowaniu;
- przesuwanie elementów na ekranie w ciekawy sposób;
- tworzenie niesamowitych gier (takich jak Cut the Rope), które działają w przeglądarce;
- przesyłanie danych między serwerem a przeglądarką;
- umożliwienie interakcji z kamerą internetową, mikrofonem i innymi urządzeniami.

Czym jest JS

JS

JavaScript tak naprawdę narodził się w połowie lat 90. XX wieku w firmie Netscape jako LiveScript (projekt Mocha) — język skryptowy rozszerzający standardowy HTML m.in. o możliwość interakcji z użytkownikiem przeglądającym stronę. Jego głównym projektantem był Brendan Eich. Nieco później doszło do porozumienia między firmami Netscape i Sun Microsystems, w wyniku którego pod koniec 1995 roku światło dzienne ujrzał JavaScript.



<https://www.youtube.com/watch?v=aX3ZABcC38>

Umieszczanie skryptu na stronie

```
<html>
<head>
  <title>Super skrypt</title>
</head>
<body>
<script>
  ...instrukcje skryptu
</script>
</body>
```

```
<script src="plik_ze_skryptem.js"></script>
```

```
<head>
  <title>Jestem super stroną</title>

  <script src="modernizr.js"></script>
  <script src="jquery.js"></script>
</head>
<body>
  ..treść strony...

  <script src="super-script.js"></script>
  <script>
    alert('Witaj świecie');
  </script>
</body>
```

Pierwszy skrypt

Wyświetlamy komunikat:

W oknie wiadomości

```
alert('Witaj wiecie!!!');
```

W konsoli

```
console.log('Witaj wiecie!!!');
```

Na stronie

```
document.write('Witaj wiecie!!!');
```

Typy danych

Typ danych to po prostu określenie rodzaju danych. Przykładowo, typ całkowitoliczbowy określa liczby całkowite.

Występujące w **JavaScriptcie** typy danych można podzielić następująco:

- typ liczbowy - *Number*,
- typ łańcuchowy - *String*,
- typ logiczny - *Boolean*,
- typ obiektowy,
- typy specjalne

<https://playcode.io/>

- typ niezdefiniowany - *Undefined*,
- typ nieokreślony - *Null*,

Przykłady

JS

```
var a = 12;  
console.log(a);  
console.log(typeof(a));
```

12
'number'

```
var a;  
console.log(a);  
console.log(typeof(a));
```

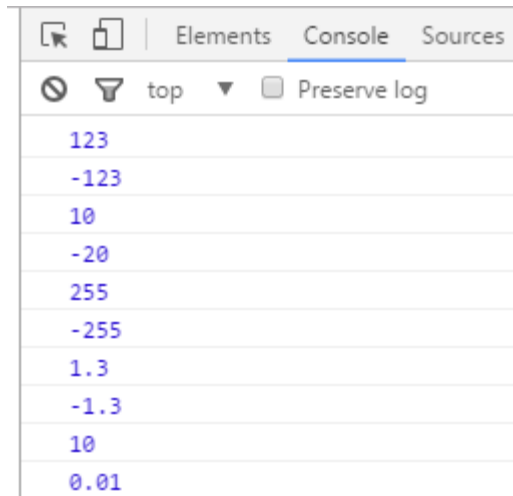
undefined
'undefined'

```
var a = null;  
console.log(a);  
console.log(typeof(a));
```

null
'object'

Przykłady literałów liczbowych:

```
<script>
  var x = 123; console.log(x);
  var x = -123; console.log(x);
  var x = 012; console.log(x);
  var x = -024; console.log(x);
  var x = 0xFF; console.log(x);
  var x = -0xFF; console.log(x);
  var x = 1.3; console.log(x);
  var x = -1.3; console.log(x);
  var x = 0.1E2; console.log(x);
  var x = 1.0E-2; console.log(x);
</script>
```



| Elements | Console | Sources |
|---|---------|---------|
| top ▼ <input type="checkbox"/> Preserve log | | |
| 123 | | |
| -123 | | |
| 10 | | |
| -20 | | |
| 255 | | |
| -255 | | |
| 1.3 | | |
| -1.3 | | |
| 10 | | |
| 0.01 | | |

Typ łańcuchowy i logiczny

```
var a = "Programuję w JavaScript";  
console.log(a);  
console.log(typeof(a));
```

'Programuję w JavaScript'
'string'

```
var a = 15 > 10;  
console.log(a);  
console.log(typeof(a));
```

true
'boolean'

```
var a = 15 == 10;  
console.log(a);  
console.log(typeof(a));
```

false
'boolean'

Konwersja na typ Number

Do konwersji na typ Number służą dwie globalne metody:

- `parseInt(wartość)` - zamieniana jest "wartość" na liczbę całkowitą.
- `parseFloat(wartość)` - zamieniana jest "wartość" na liczbę zmiennoprzecinkową.

```
<body>
  <script>
    var a = prompt('Podaj liczbę a:', '');
    var b = prompt('Podaj liczbę b:', '');
    var suma = a + b;

    document.write('Suma: ' + a + '+' + b + '=' + suma);
  </script>
</body>
```

Nazwy zmiennych i co z tym var?

```
var myNumber;  
var myFavoriteNumber;  
var my_favorite_number;
```

ES2015

„let” oraz „const”, to słowa kluczowe, które zostały wprowadzone do języka JavaScript dzięki nowemu standardowi ECMAScript- ES6 (został później przemianowany na ES2015). „var”, „let” oraz „const” służą do tego samego, bo dzięki nim, możemy zdefiniować zmienną. Jednak, jak to zwykle w życiu bywa, diabeł tkwi w szczegółach.

Operatory matematyczne

Poniżej zamieszczam przykłady dla $y = 5$

| Operator | Nazwa działania | Równanie | Wynik | |
|----------|-----------------------------------|--------------|---------|------------|
| + | Dodawanie | $x = y + 2$ | $y = 5$ | $x = 7$ |
| - | Odejmowanie | $x = y - 2$ | $y = 5$ | $x = 3$ |
| * | Mnożenie | $x = y * 2$ | $y = 5$ | $x = 10$ |
| / | Dzielenie | $x = y / 2$ | $y = 5$ | $x = 2.5$ |
| % | Reszta z dzielenia | $x = y \% 2$ | $y = 5$ | $x = 1$ |
| ** | Potęgowanie (to samo co Math.pow) | $x = y ** y$ | $y = 5$ | $x = 3125$ |
| ++ | Inkrementacja | $x = ++y$ | $y = 6$ | $x = 6$ |
| | | $x = y++$ | $y = 6$ | $x = 5$ |
| -- | Dekrementacja | $x = --y$ | $y = 4$ | $x = 4$ |
| | | $x = y--$ | $y = 4$ | $x = 5$ |

Operatory porównania

przykłady dla **x = 5**

| Operator | Opis | Równanie | Zwróci |
|--------------|--|-----------|--------|
| == | równe | x == 8 | false |
| != | różne | x != 8 | true |
| === | równa wartość i taki sam typ danychW naszym przykładzie x to numer, a 5 po prawej stronie jest tekstem | x === 5 | true |
| | | x === "5" | false |
| !== | różne wartości lub różny typ danychW naszym przykładzie x to numer, a 5 po prawej stronie jest tekstem | x !== "5" | true |
| | | x !== 5 | false |
| > | większe od | x > 8 | false |
| < | mniejsze od | x < 8 | true |
| >= | większe bądź równe od | x >= 8 | false |
| <= | mniejsze bądź równe od | x <= 8 | true |

Operatory logiczne

dla $x = 6$ i $y = 3$:

| Operator | Opis | Przykład | Wynik |
|-------------------------|---|--|---|
| <code>&&</code> | and (i) | <code>(x < 10 && y > 1)</code> | Prawda, bo x jest mniejsze od 10 i y jest większe od 1 |
| <code> </code> | or (lub) | <code>(x > 8 y > 1)</code> | Prawda, bo x nie jest większe od 8, ale y jest większe od 1 |
| <code>^</code> | xor (jeden z, ale nie dwa równocześnie) | <code>(x === 6 ^ y === 3)</code> | Fałsz, bo obydwa są prawdziwe |
| <code>!</code> | not (negacja) | <code>!(x === y)</code> | Prawda, bo negujemy to, że <code>x === y</code> |

Operatory przypisania

dla $x = 10$ i $y = 5$:

| Operator | Przykład | Równoznaczne z | Wynik |
|----------|------------|----------------|----------|
| = | $x = y$ | $x = y$ | $x = 5$ |
| += | $x += y$ | $x = x + y$ | $x = 15$ |
| -= | $x -= y$ | $x = x - y$ | $x = 5$ |
| *= | $x *= y$ | $x = x * y$ | $x = 50$ |
| /= | $x /= y$ | $x = x / y$ | $x = 2$ |
| %= | $x \% = y$ | $x = x \% y$ | $x = 0$ |

Obiekt Math()

Właściwości:

Javascript udostępnia nam obiekt **Math**. Obiekt ten za pomocą swoich funkcjonalności ułatwia nam przeprowadzanie operacji matematycznych.

| | | |
|---------------------|--|--------------------|
| Math.E | - Zwraca stałą Eulera, która wynosi ok. 2.71 | 2.718281828459045 |
| Math.LN2 | - Zwraca logarytm dwóch, tj. ok. 0.69 | 0.6931471805599453 |
| Math.LN10 | - Zwraca logarytm z dziesięciu, tj. ok. 2.30 | 2.302585092994046 |
| Math.LOG2E | - Zwraca logarytm o podstawie 2 z liczby E, czyli ok. 1.44 | 1.4426950408889634 |
| Math.LOG10E | - Zwraca logarytm o podstawie 10 z E, czyli ok. 0.43 | 0.4342944819032518 |
| Math.PI | - Zwraca wartość liczby Pi, czyli ok. 3.14 | 3.141592653589793 |
| Math.SQRT1_2 | - Zwraca pierwiastek kwadratowy z 0.5, czyli ok. 0.70 | 0.7071067811865476 |
| Math.SQRT2 | - Zwraca pierwiastek kwadratowy z 2, czyli ok. 1.41 | 1.4142135623730951 |

Obiekt Math()

Metody:

Javascript udostępnia nam obiekt **Math**. Obiekt ten za pomocą swoich funkcjonalności ułatwia nam przeprowadzanie operacji matematycznych.

| | |
|---------------------------|--|
| Math.abs(liczba) | - Zwraca wartość absolutną liczby |
| Math.acos(liczba) | - Zwraca arcus cosinus z liczby (podanej w radianach) |
| Math.asin(liczba) | - Zwraca arcus sinus z liczby (podanej w radianach) |
| Math.atan(liczba) | - Zwraca arcus tangens z liczby (podanej w radianach) |
| Math.ceil(liczba) | - Zwraca najmniejszą liczbę całkowitą, większą lub równą podanej liczbie |
| Math.cos(liczba) | - Zwraca cosinus liczby (podanej w radianach) |
| Math.exp(liczba) | - Zwraca wartość E podniesionej do potęgi wyrażonej podanym argumentem |
| Math.floor(liczba) | - Zwraca największą liczbę całkowitą mniejszą lub równą podanej liczbie |

Obiekt Math()

Metody:

Javascript udostępnia nam obiekt **Math**. Obiekt ten za pomocą swoich funkcjonalności ułatwia nam przeprowadzanie operacji matematycznych.

| | |
|-----------------------------------|---|
| Math.log(liczba) | - Zwraca logarytm naturalny liczby |
| Math.max(liczba1, liczba2) | - Zwraca większą z dwóch liczb |
| Math.min(liczba1, liczba2) | - Zwraca mniejszą z dwóch liczb |
| Math.pow(liczba1, liczba2) | - Zwraca wartość liczby1 podniesionej do potęgi liczby2 |
| Math.random() | - Zwraca wartość pseudolosową z przedziału 0 - 1 |
| Math.round(liczba) | - Zwraca zaokrąglenie danej liczby do najbliższej liczby całkowitej |
| Math.sin(liczba) | - Zwraca sinus liczby (podanej w radianach) |
| Math.sqrt(liczba) | - Zwraca pierwiastek kwadratowy liczby |
| Math.tan(liczba) | - Zwraca tangens liczby (podanej w radianach) |