

JavaScript :: Lectia 1

1. Introducere

JavaScript (pe scurt "JS") a fost inventat de Brendan Eich co-fondator al Mozilla Project, Mozilla Foundation si Mozilla Corporation. JS este in primul rand un limbaj client-side, adica este interpretat si executat de browserul utilizatorului. El a fost inventat pentru a oferi paginilor web interactivitate.

2. Unde inseram codul JavaScript?

Codul JS poate fi inserat in interiorul sectiunii `<head>` sau in `<body>`.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Document</title>
    <!-- javascript code -->
</head>
<body>
    <!-- page content -->
    <!-- javascript code -->
</body>
</html>
```

Codul JS este de doua feluri: intern si extern. Codul intern apare intre tagurile `<script></script>`, iar cel extern este inclus in pagina folosind:

```
<script src="path/to/our/script/filename.js"></script>
```

Daca in tagul `<script>` apare atributul `src` atunci codul JS din interiorul tagului este ignorat.

```
<script src="script.js">
    alert('message'); // aceasta instructiune este ignorata!!
</script>
```

Pentru ca aceasta instructiune sa nu mai fie ignorata se imparte codul in doua:

```
<script src="script.js"></script>
<script>
    alert('message');
</script>
```

Pentru a include in fisierul nostru mai multe fisiere JS vom folosi mai multe taguri `<script>`:

```
<script src="global.js"></script>
<script src="jquery.js"></script>
```

3. Unde se afiseaza datele din JavaScript?

Folosind limbajul de programare JavaScript putem:

- afisa mesaje in consola browser-ului. Acest gen de mesaje sunt folosite pentru debugging:

```
console.log('mesaj');
console.info('mesaj');
console.warn('mesaj');
console.error('mesaj');

var data = [
    {firstname: 'John', surname: 'Doe'},
    {firstname: 'Jane', surname: 'Doe'}
];
console.table(data);

console.time('timerId'); // porneste timer-ul
console.timeEnd('timerId') // afiseaza valoarea timer-ului
```

- adauga / modifica / sterge un element din pagina

- afisa o alerta pe ecran folosind `alert('mesaj')`

- afisa o confirmare (o fereastra care contine butoanele `cancel` si `ok`) pe ecran folosind `confirm('mesaj')`

- afisa o fereastră care contine un mesaj si un camp text in care i se solicita userului o valoare folosind `prompt('mesaj')`

- solicita informatii de la server si a le afisa in pagina sau putem solicita serverului sa realizeze anumite comenzi fara a reincarca pagina, folosind tehnologia ajax (Asynchronous JavaScript And XML)

- si multe altele

4. Sintaxa limbajului

4.0 Standarde de codare

<http://jstherightway.org/>

<https://github.com/rwaldron/idiomatic.js>

<https://contribute.jquery.org/style-guide/>

<https://google.github.io/styleguide/javascriptguide.xml>

4.1 Variabile (var vs let)

a. Denumirea variabilelor:

1. Numele variabilelor nu pot sa inceapa cu o cifra, dar poate contine cifra in interiorul sau la finalul numelui.
2. Numele variabilelor nu pot sa contina caractere speciale: +, -, /, *, \, ^, !, #, :, ', etc. Exceptie face '_' (underscore-ul). El poate sa apara la inceput, in interiorul sau la sfarsitul numelui variabilei. De asemenea numele variabilelor pot sa inceapa cu \$.
3. Numele variabilelor nu pot sa contina semne de punctuatie.
4. Numele variabilelor nu pot sa contina spatiu. Niciodata!
5. Nu putem sa folosim ca nume de variabile cuvinte cheie din JavaScript, precum `window`, `location`, `string`, `open`, etc. Dar putem sa le combinam cu alte cuvinte precum `thatWindow`, `someString`, etc.
6. Numele variabilelor in JavaScript sunt *case-sensitive*! (`myVariable`, `MyVariable`, `myvariable` sunt 3 variabile diferite!)

b. Tipuri de variabile:

1. Numeric (int, float, double):

```
var age = 25;
var pi = 3.14

var price1 = 23;
var price2 = 19;
var total = price1 + price2; // = 23 + 19 = 42

var x = 1 + 2 + 3;
```

Funcții:

```
var a = 3.14;
console.log(a.toFixed(3)); // afiseaza 3.140 in consola
console.log(a.toFixed(1)); // afiseaza 3.1 in consola
```

Obiectul Math:

```
var pi = Math.pi; // returneaza valoare lui PI
var x = Math.sqrt(4); // returneaza radical din 4 (= 2)

Math.ceil(x);
Math.floor(x);
Math.round(x);
Math.max(x, y, z, ..., n);
Math.min(x, y, z, ..., n);
Math.pow(x, y);
```

2. Text (string)

```
var volunteerName = 'John Doe';
var fullName = 'John' + ' ' + 'Doe';
var a = "4" + 1 + 6; // a = "416"
var b = 5 + 2 + "3"; // b = "73"
var data = "John's first trip";
```

Best practices

- string-urile pot sa fie scrise intre " sau '. Nu este nicio diferenta intre modalitatile in care le interpreteaza JS-ul. Dar **FOARTE IMPORTANT** este sa ne alegem un delimitator si sa il folosim doar pe acela!!!

Funcții:

```
'John Doe'.length // returneaza lungimea string-ului 'John Doe'
(numarul de caractere din string)

'John Doe'.charAt(n) // returneaza caracterul de pe pozitia n

'John '.concat('Doe') // returneaza 'John Doe'

'John Doe'.indexOf('D') // returneaza pozitia lui 'D' (5)

'John Doe'.lastIndexOf('o') // returneaza ultima pozitie a lui 'o'
(6)

'John'.repeat(3) // returneaza 'JohnJohnJohn'

'John Doe'.replace('John', 'Jane') // inlocuieste 'John' cu 'Jane'
si returneaza 'Jane Doe'

.toLowerCase()
.toUpperCase()
.trim()

.substr(index, length)
.substring(indexStart, indexEnd)
.split([separator, limit])
```

3. Boolean

```
var isAccepted = true;
var myFalseVar = false;
var isMajor = age >= 18;
```

4. Siruri (array)

```
var names = ['John', 'Jane', 'Jack'];
var ages = [23, 28, 25];
var emptyArray = [];

console.log(names[0]); // afiseaza 'John' in consola
console.log(ages[2]); // afiseaza 25 in consola
```

Funcții:

```
.filter()
```

.find()

.indexOf() - returneaza pozitia unui element intr-un array.

Cautarea se face tinand cont de literele mari si mici (case sensitive)

```
var destinations = ['Fiji', 'Peru', 'Nepal', 'Ghana'];
console.log(destinations.indexOf('Argentina'));
    // returneaza -1 pentru ca nu exista elementul
    'Argentina' in array-ul destinations
console.log(destinations.indexOf('Fiji'));
    // returneaza 0 pentru ca 'Fiji' se afla pe prima
    pozitie din array
```

.lastIndexOf() - cauta un element intr-un array incepand de la sfarsit spre inceput; deci returneaza ultima pozitie pe care se afla acesta. Cautarea este case sensitive

```
var projects = ['Care', 'Sports', 'Conservation', 'Care',
'Building'];
console.log(projects.lastIndexOf('Medicine'));
    // returneaza -1 pentru ca 'Medicine' nu se afla in
    array-ul projects
console.log(projects.lastIndexOf('Care'));
    // returneaza 3 pentru ca ultima aparitie a
    elementului 'Care' este pe pozitia 3
```

.join() - concateneaza elementele unui array folosind un delimitator specificat ca parametru pentru functie

```
var destinations = ['Peru', 'Ghana', 'Argentina'];
console.log(destinations.join());
    // afiseaza 'Peru,Ghana,Argentina' in consola

console.log(destinations.join(' and '));
    // afiseaza 'Peru and Ghana and Argentina' in
    consola
```

.map()

.pop() - scoate si returneaza ultimul element din array

```
var data = ['A', 'B', 'C'];
```

```
console.log(data.pop()); // afiseaza 'C'  
console.log(data); // afiseaza ['A', 'B']
```

.push() - adauga unul sau mai multe elemente la finalul array-ului si returneaza noua lungime a lui.

```
var data = ['X'];  
data.push('A');  
data.push('B', 'C');  
console.log(data); // afiseaza ['X', 'A', 'B', 'C']
```

.shift()
.slice()
.sort()
.unshift()