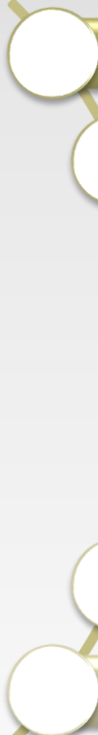


Introducere în javascript

DĂM VIAȚĂ DOCUMENTELOR CU LIMBAJUL DE PROGRAMARE JAVASCRIPT

Cuprins



Introducere
Inserare JS in HTML
Obiecte in JS
Tipuri de date
Variabile si Operatori
Display
Funcții

Introducere

HTML este pentru a localiza documentele și pentru a le afișa pe ecran;



Documentele create cu HTML sunt statice;



Prin utilizarea JavaScript putem da viață documentelor:

Interacționăm cu utilizatorul prin intermediul butoanelor, formularelor sau jocurilor

Creează efecte vizuale

Modifică modul de redare a documentului în fereastra navigatorului

Controlează fereastra navigatorului

Introducere

JavaScript este o limbaj dinamic multi-paradigmă, cu **tipuri și operatori**, **obiecte standard incluse și metode**

Instrucțiunile în JavaScript seamănă foarte mult cu Java și C

Mici blocuri de instrucțiuni JavaScript sunt folosite pentru a adăuga funcționalitate paginii HTML

Comentarii JS pentru o linie
//comentariu și /* */ pentru un bloc de instrucțiuni

Introducere

Codul JavaScript este inclus în documentul HTML astfel:

Plasare în HEAD folosind etichetele `<script>` și `</script>`

Plasare în BODY, în locul unde trebuie executat și pus sub aceleași etichete `<script>` și `</script>`

Inclus într-un fișier separat salvat cu extensia .js și apelat în HEAD

`<script src="myscript.js">`

Poziționarea codului JS în HEAD

```
<!DOCTYPE html>
<html><head>
<script>
function myFunction() {
  document.getElementById("demo").innerHTML = „Ai reușit
să schimbi paragraful doar cu un click.";
}
</script>
</head>
<body>

<h1>Prima pagina cu JS</h1>
<p id="demo">Pot schimba o linie din pagina HTML cu un
click</p>
<button type="button" onclick="myFunction()">Încearcă să
schimbi</button>

</body>
</html>
```

Prima pagina cu JS

Pot schimba o linie din pagina HTML cu un click

Încearcă să schimbi

Prima pagina cu JS

Ai reușit să schimbi paragraful doar cu un click

Încearcă să schimbi

Poziționarea codului JS în Body

```
<!DOCTYPE html>

<html>

<body>

<h2>Am început să învăț</h2>

<button type="button"

onclick="document.getElementById('demo').innerHTML = Date()">

Click pentru a afișa DATA.</button>

<p id="demo"></p>

</body>

</html>
```

Am început să învăț

Click pentru a afișa DATA.

Am început să învăț

Click pentru a afișa DATA.

Sat Mar 23 2019 20:33:20 GMT+0200 (Ora standard a Europei de Est)

Poziționarea codului JS în EXTERN

Fișier Script.js

```
function myFunction() {  
  document.getElementById("demo").innerHTML = „ Ai reușit să  
  schimbi paragraful doar cu un click  
  ”;  
}
```

```
<!DOCTYPE html>  
<html> <head><script src="Script.js"></script>  
</head>  
<body>  
  <h1>Prima pagina cu JS</h1>  
  <p id="demo">Pot schimba o linie din pagina HTML cu un  
  click</p>  
  <button type="button" onclick="myFunction()"> Încearcă  
  să schimbi</button>  
</body>  
</html>
```


Obiecte în JavaScript

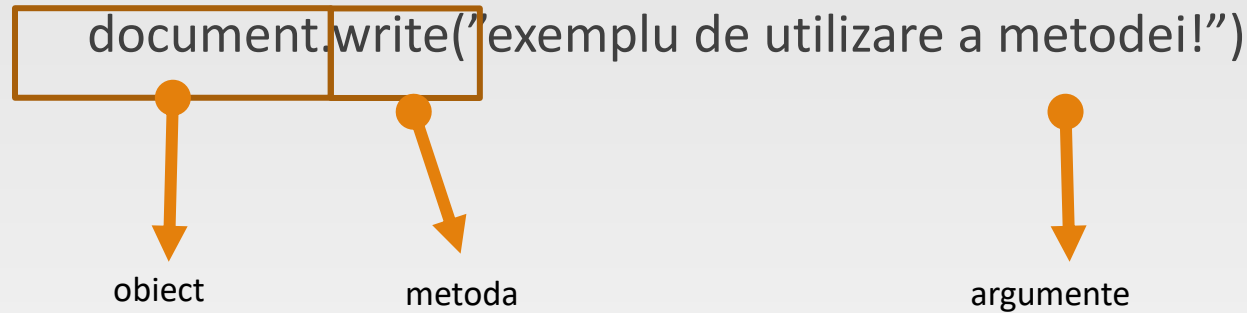
Obiectele sunt secvențe de cod program și date care pot fi tratate ca un tot unitar

Liniile individuale de program se numesc **declarații**

Grupurile de declarații înrudite, asociate unui obiect se numesc **metode**.

Unul din obiectele cele mai importante este obiectul **Document**, care reprezintă documentul afișat în ferestrea navigatorului.

Obiecte în JavaScript



`document.getElementById("demo").innerHTML = „Pot schimba cu un click.”`

Metoda de afișare în JavaScript

Scrierea într - un element HTML, folosind **innerHTML**;

```
<p> Ce imi afiseaza innerHTML </p>
```

```
<p id="demo1">...11....</p>
```

```
<script>
```

```
document.getElementById("demo1").innerHTML = 5 + 6;
```

```
</script>
```

Cum afișez în JavaScript

Scrierea în ieșirea HTML folosind
document.write();

```
<!DOCTYPE html>
```

```
<html><body>
```

```
<h2>Prima afișare cu JS</h2>
```

```
<button type="button"  
onclick="document.write(4+9)">Try it</button>
```

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

Prima afișare cu JS

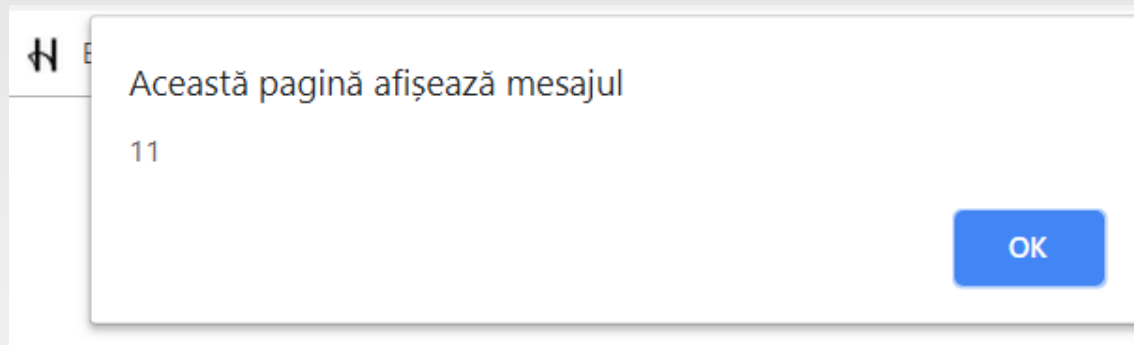
Try it

11

Cum afișez în JavaScript

Scrierea într - o casetă de avertizare, folosind **window.alert();**

```
<script>  
window.alert(5 + 6);  
</script>
```



Scrierea în consolă browser - ul, folosind **console.log();**

Tipuri de date

Tipurile JS sunt:

1. Numere `var x = 16; var X = 17;`
2. String-uri `var y = "16+17";`
3. Boolean `true` sau `false`
4. Obiecte `var x = {firstName:"John", lastName:"Doe"};`
 - a) Funcții
 - b) Array
 - c) Date
 - d) RegExp
5. Undefined
6. Null

Concept - tipuri de date

Dacă adăugați la o variabilă un număr și un șir de caractere, JS va trata numărul ca un șir de caractere

Exemplu

```
var x = 621 + "AD";
```

621AD

```
var y = 621 + 622 + "AD";
```

1243AD

```
var z = "AD" + 621 + 622;
```

AD621622

Concept - tipuri de date

Tipurile de date sunt dinamice

```
var x;
```

```
x = 622;
```

```
x = "AD";
```

Un șir de caractere poate fi introdus cu "....." sau '.....'

```
var nume = "Popescu";
```

```
var prenume = 'Maria';
```

Array este o matrice

```
var cars = ["Saab", "Volvo", "BMW"];
```


Concept - tipuri de date

Numerele pot fi scrise întregi sau cu zecimale

```
var x1 = 14.00;  
var x2 = 14;  
  
var y = 234e5;    // 23400000  
var z = 123e-5;   // 0.00123
```

Booleans: true sau false

```
var x = 5;  
var y = 5;  
var z = 6;  
(x == y)    // Returns true  
(x == z)    // Returns false
```

Concept - tipuri de date

Undefined

var x; // este o variabilă cu **valoare nedefinită** și cu **tip nedefinit**

var x="" //este o variabilă cu **valoarea ""** și cu **tip string**

Null

```
<script>
```

```
var person = {nume:"Popescu", prenume:"Ion"};
```

Nu o să îmi afișeze nimic

```
//person = null;
```

```
document.getElementById("demo").innerHTML = person;
```

```
</script>
```

Concept tipuri de date

Pentru a putea observa ce tip este o dată folosesc următoarea instrucțiune:

typeof

Poate să îmi afișeze tipurile de date primitive: numere, string-uri, booleans și tipuri de date complexe: obiect și funcție

```
typeof " Popescu"      // Returns "string"  
typeof 622              // Returns "number"  
typeof true            // Returns "boolean"
```

Concept tipuri de date

typeof {name:'John', age:34}

typeof [1,2,3,4] // Returnează "object"

 // Returnează "object"

typeof null

 // Returnează "object"

typeof function myFunc(){ *ce dorim să facă funcția;* }

 // Returnează "function"

Variabile

Declarăm folosind **var;**

Sunt folosite pentru a stoca date;

Numele unei variabile trebuie să înceapă cu **o literă**, sau **\$** sau cu **_**

Nu au TIP

Sunt 2 tipuri de variabile

GLOBALE

LOCALE

Variabile

Variabilele se pot declara multiplu pe o linie sau mai multe

```
var x=5 , nume="Popescu", vârsta=15;
```

Sau

```
var x=5;
```

```
var nume="Popescu";
```

```
var vârsta=15;
```

Operatori

Operatori aritmetici:

+ - * / % ++ -- **

Operatori de comparație

< <= == != >= >

Operatori logici

&& || !

Operatori de atribuire

+= -= *= /= %= <<= >>=

X-=y x=x-y

Operatori

Operator la nivel de string-uri: +

```
var x="Popescu" + " " + "Vasile"
```

Operatori pentru egalitate și diferit:

== **și** **!=**

```
var x=5, y=2;
```

```
y=y+x=7;
```

```
y+=x;
```


Funcții

Permit gruparea declarațiilor în unități logice ce pot fi folosite în program pentru executarea unor operații;

Funcțiile se includ între etichetele **<script>** și **</script>**;

Funcția trebuie **declarată** înainte de a fi folosită, după sintaxa

```
function nume_funcție (listă argumente) { declarații }
```

```
function adunare (x,y) { return x+y; }
```

Funcții

```
<!DOCTYPE html>
```

```
<html><body>
```

```
<h2>JavaScript Functions</h2>
```

```
<p id="demo"></p>
```

```
<script>
```

```
function inmultire(p1, p2) {
```

```
var p3=p1*p2;
```

```
return p3;
```

```
}
```

```
document.getElementById("demo").innerHTML = inmultire(4, 3);
```

```
</script>
```

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

4*3=12

Funcția retur

Va opri executarea funcției și va returna variabila indicată

```
Const a = 7;
```

```
function operații_artimetice(a, b) {  
  var c = a + a * b + b;  
  return c;  
}
```

Apelare: `function` operații_artimetice(4, 5)

Var b = `operatii_aritmetice`(4,5) = 29

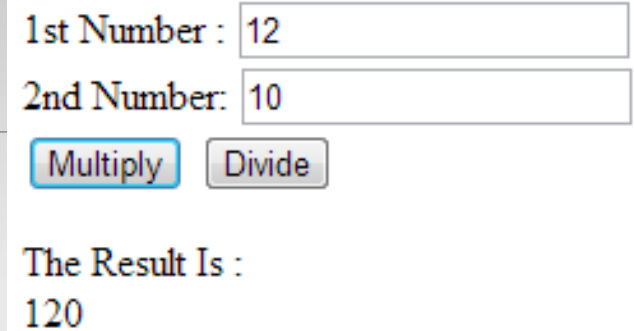
Var a + b = 7+29

Rezultat: 29

Exercițiu

```
<form>
1st Number: <input type="text" id="firstNumber"/><br>
2nd Number: <input type="text" id="secondNumber" /><br>
<input type="button" onClick="multiplyBy()" Value="Multiply"/>
<input type="button" onClick="divideBy()" Value="Divide"/>
</form>
<p>The Result is : <br>
<span id = "result"></span>
</p>
```

```
function multiplyBy() {
    num1 = document.getElementById("firstNumber").value;
    num2 = document.getElementById("secondNumber").value;
    document.getElementById("result").innerHTML = num1 * num2;
}
```



A screenshot of a web form. It contains two text input fields: '1st Number' with the value '12' and '2nd Number' with the value '10'. Below these are two buttons: 'Multiply' (highlighted with a blue border) and 'Divide'. Below the buttons, the text 'The Result Is :' is followed by the value '120'.

Functii matematice speciale

`Math.max()` – afiseaza maximul

`Math.min()` – afiseaza minimul

`Math.random()` – afiseaza un numar random intre 0 si 1

`Math.exp()` – returneaza e^x , unde x este argument

`Math.pow()` – returneaza baza la exponent

Functii string

`.length` – afiseaza lungimea string-ului

`indexOf()` – returneaza poziti de inceput a string-ului

`lastIndexOf()` – returneaza ultima pozitie a string-ului

`Search()` – cauta un string in alt string si afiseaza pozitia de inceput

`Substring(str, position)` – sterge dintr-un string litera de pe pozitia specificata

Exercitiu

Sa se elimine litera specificata din cuvant.

```
function remove_character(str, char_pos) {  
    part1 = str.substring(0, char_pos);  
    part2 = str.substring(char_pos + 1,  
str.length);  
    return (part1 + part2);  
}  
console.log(remove_character("Student",3));
```

Exercitiu

Scriveți un program JavaScript pentru a crea un nou șir de caractere dintr-un șir dat, luând ultimele 3 caractere și adăugându-le atât în față cât și în spate. Lungimea șirului trebuie să fie mai mare sau egală cu 3.

```
function front_back3(str) {  
  if (str.length >= 3) {  
    str_len = 3;  
    back = str.substring(str.length - 3);  
    return back + str + back;  
  } else  
    return false;  
}  
console.log(front_back3("abcd"));
```

bcd**abcd**bcd

Funcția if

Instrucțiunea **If**:

if (*condiție*) { *ce vreau sa returnez sau sa afișez*};

if (*condition*) {*ce vreau sa returnez sau sa afișez*};
 else {*ce vreau sa returnez sau sa afișez*};

Funcția if

```
<script>
```

```
var x = 5, y=2;
```

```
y += x;
```

```
if (x==y) {document.write("Adevărat");}  
    else { document.write("false");}
```

```
document.getElementById("demo").innerH  
TML =y;
```

```
</script>
```

Ce afișează funcția:
7
false

Exercitiu

Scriveți un program JavaScript pentru a calcula diferența absolută dintre un număr specificat și 19. Returnează tripla diferență absolută a acestora dacă numărul specificat este mai mare decât 19.

```
function diff_num(n) {  
    if (n <= 19) { return (19 - n);  
    }  
    else {  
        return (n - 19) * 3;  
    }  
}  
  
console.log(diff_num(12));
```

Funcții repetitive

Instrucțiunea while:

```
while (condiție) { // codul care va fi executat  
}
```

Instrucțiunea for:

```
for (condiție 1; condiție 2 ; incrementare; ) {  
    // codul care va fi executat  
}
```

Funcția while

```
<script>
```

```
var text = "";
```

```
var i = 0;
```

```
while (i < 5) {
```

```
    text += "<br>The number is " + i;
```

```
    i++;
```

```
}
```

```
document.getElementById("demo").innerHTML = text;
```

```
</script>
```

The number is 0

The number is 1

The number is 2

The number is 3

The number is 4

Funcția for

```
<script>
```

```
var text = "";
```

```
var i;
```

```
for (i = 0; i < 5; i++) {
```

```
    text += "The number is " + i + "<br>";
```

```
}
```

```
document.getElementById("demo").innerHTML = text;
```

```
</script>
```

The number is 0

The number is 1

The number is 2

The number is 3

The number is 4

Funcția switch

```
switch(expresie) {  
    case x:  
        // code block  
        break;  
    case y:  
        // code block  
        break;  
    default:  
        // code block  
}
```

```
<!DOCTYPE html>
<html><body>
<p id="demo"></p>
```

```
<script>
var x = 0;
switch (x) {
  case 0:
    text = "Off";
    break;
  case 1:
    text = "On";
    break;
  default:
    text = "No value found";
}
document.getElementById("demo").innerHTML = text;
</script>
</body></html>
```


Exercitii

1. Ghiceste viitorul:

- ❖ Stocați următoarele în variabile: numărul de copii, numele partenerului, locația geografică, locul de muncă.
- ❖ Atunci când apăsați butonul **Ghiceste viitorul** să se afișeze următorul mesaj "Vei ajunge un X (locul de muncă) în locația Y, căsătorit cu Z și vei avea N copii."

Exerciții

2. Să se realizeze o pagina HTML cu o listă sortată cu litere romane cu următoarele elemente:

- Animale
- Păsări
- Insecte
- Orice vietate

Să se scrie un script intern documentului HTML prin care se va schimba ultimul obiect din lista – Orice vietate – cu Fluturi prin apăsarea unui buton cu eticheta *Înlocuiește*.

Link-uri utile

http://proeco.uv.ro/lectii/html/14_aplicatii_javascript.htm

