

JavaScript

Limbajul JavaScript

JavaScript este un limbaj care se executa pe browser. In JavaScript, putem implementa, printre altele, si urmatoarele functionalitati :

- Ceasuri si timere
- Actiuni asupra cursorului si miscarii mouseului
- Meniuri drop-down
- Mesaje si alerte
- Ferestre pop-up
- Validarea formularelor

Codul JavaScript este localizat in head-ul paginii. Pentru a incarca un cod JavaScript, fie declaram functiile intre tagurile `<script></script>`, fie incarcam un fisier *.js in care avem definite acele functii. Pentru a apela functiile JavaScript, introducem tagul `<script>` acolo unde este nevoie, in body.

```
<html>
<head>
  <script type="text/JavaScript">
    <!--
      function numeFunctie() {
        ...
      }
    //-->
  </script>
</head>
<body>
</body>
</html>
```

Pentru a incarca fisiere externe cu cod JavaScript, folosim urmatoarea linie de cod :

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

Operatorii in JavaScript sunt : +(adunare), - (scadere), * (inmultire), / (impartire), % (modul) .

Operatorii de comparatie sunt : == (egal), != (diferit), < (mai mic), >(mai mare), <= (mai mic sau egal), >=(mai mare sau egal).

Referirea la pagina curenta se face prin cuvantul cheie **document**. Spre exemplu, pentru a scrie ceva pe pagina, vom folosi **document.write()**.

Variabile

O variabila este un simbol care reprezinta anumite informatii setate de programator. Variabilele se utilizeaza in acelasi mod ca in majoritatea limbajelor de programare. Diferenta este ca, in JavaScript, acestea nu trebuie declarate inainte de a le folosi. Totusi, se recomanda declararea lor, folosind termenul **var**.

```
<script type="text/JavaScript">
    <!--
    var linieNoua = "<br />"
    var variabila = "Salutari!"

    document.write(variabila)
    document.write(linieNoua)
    //-->
</script>
```

Functii

Functiile sunt bucati de cod care nu se executa decat atunci cand sunt apelate. Ele sunt utile si pentru ca ajuta la evitarea scrierii aceleiasi parti de cod de mai multe ori.

```
<script type="text/JavaScript">
    <!--
    function salutare() {
        alert("Salut!")
    }
    //-->
</script>
```

Funcțiile pot fi apelate în pagină, sau la anumite evenimente: click-ul pe un buton (onclick), modificarea valorii unui input (onchange), atunci când trecem cu mouseul deasupra controlului (onmouseover) etc.

```
<input type="button" value="Click!" onclick="salutare()">
```

Instructiuni

Instructiunea IF, IF/ELSE

Instructiunea IF verifică dacă o condiție este adevărată sau falsă. Dacă este adevărată, se execută instrucțiunile din prima acoladă. Dacă este falsă, atunci se execută ramura **else**.

```
if (variabila == 5){
    document.write("Este 5");
}
else {
    document.write("Nu este 5");
}
```

Putem avea mai multe condiții, folosind, astfel, instrucțiunea **else if**.

```
if (variabila == 5){
    document.write("Este 5");
}
else if (variabila == 3) {
    document.write("Este 3");
}
else {
    document.write("Nu este nici 3, nici 5");
}
```

Instructiunea WHILE

Instructiunea WHILE este compusă din două părți: verificarea condiției și executarea instrucțiunilor, atâta timp cât condiția este adevărată. Pentru a evita blocarea browserului, trebuie să fim atenți să modificăm variabila din condiție, astfel încât instrucțiunea să se întrerupă.

```
while(contor < 5){
    document.write("Contor = " + contor + "<br/>");
    contor++;
}
```

Prin operatorul ++, am crescut variabila contor cu o unitate.

Instructiunea FOR

Instructiunea FOR foloseste o variabila, pe care o initializeaza, o supune unei conditii si careia ii specifica pasul de modificare. Atunci cand valoarea variabilei nu mai respecta conditia, instructiunea nu se mai executa.

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

Comentariile pe o linie se realizeaza punand semnul `//` la inceputul liniei, iar comentariile pe mai multe linii se realizeaza punand `/*` la inceputul blocului de text comentat si `*/` la finalul acestuia.

Tablouri (Array)

Spre deosebire de variabilele obisnuite, in JavaScript, tablourile se declara astfel:

```
var unArray = new Array();  
unArray[0] = "a";  
unArray[1] = "ab";  
unArray[2] = "abc";
```

Dialoguri Javascript

Avem posibilitatea de a afisa anumite mesaje utilizatorului.

Alerta: afiseaza o anumita informatie si asteapta ca utilizatorul sa apese butonul OK

```
alert("Bine ati venit pe site! ")
```

Confirmare: afiseaza o intrebare si asteapta confirmare/infirmare de la utilizator

```
var raspuns=confirm("Vreti sa stergeti inregistrarea?")
```

Prompt: asteapta o valoare de la utilizator

```
var nume = prompt("Care este numele tau?", "")
```

Redirectare: trimite browserul la o alta adresa

```
window.location = "http://www.google.ro/"
```

Popup: deschide o fereastră nouă

```
window.open( "http://www.google.ro/", "titluPopup", "status = 1, height = 300,  
width = 300, resizable = 0" )
```

Pentru a deschide o fereastră simplă, cu dimensiunile default ale browserului, putem apela:

```
window.open( "http://www.google.ro/" )
```

Validarea formurilor

Pentru ca se executa la nivelul browserului, JavaScript este foarte util in validarea datelor introduse in formulare. Astfel, putem verifica daca:

- S-a introdus ceva intr-un input text
- Textul dintr-un input este numeric
- Textul dintr-un input e format doar din litere
- Textul dintr-un input este format doar din litere si cifre
- Textul dintr-un input are o anumita lungime
- S-a selectat o valoare in select
- S-au completat toate informatiile obligatorii din formular

Pentru a ne accesa proprietatile unui anumit camp (definit print-un **ID**) din formular, putem folosi functia **document.getElementById('idControl')**. Pentru a lua valoarea campului, procedam astfel:

```
var valoare=document.getElementById("idControl").value;
```

Pentru a afla daca s-a scris un text in control, verificam lungimea acestuia:

```
if(valoare.length == 0)
```

Daca vrem sa verificam ca textul corespunde unui anumit tipar (doar cifre, cifre si litere, etc), folosim expresii regulate.

```
var doarCifre = /^[0-9]+$/;  
var doarLitere = /^[a-zA-Z]+$/;  
var cifreSiLitere = /^[0-9a-zA-Z]+$/;  
if(valoare.match(doarCifre))  
    alert("Doar cifre");  
else if (valoare.match(doarLitere))  
    alert("Doar litere");  
else if (valoare.match(cifreSiLitere))  
    alert("Cifre si litere");  
else alert("Nu corespunde niciunui tipar");
```

Verificarea lungimii textului (stringului):

```
if (valoare.length<limitaMinima)  
    alert("Textul nu are suficiente caractere");
```

Validarea unei adrese de email presupune folosirea unei expresii regulate:

```
var email = /^[\\w\\-\\.\\+]+@[a-zA-Z0-9\\.\\-]+\\. [a-zA-z0-9]{2,4}$/;
```

Funcțiile de validare pentru un control se apelează când se schimbă valoarea controlului care trebuie validat (`onchange=numefuncție()`). Există posibilitatea de a valida întreg formularul deodată. În momentul în care se face validarea, funcția va returna `true`, dacă toate câmpurile sunt valide sau `false`, dacă cel puțin un câmp este invalid. Validarea întregului formular se face prin două metode:

1. La metoda `onsubmit`, a formularului. Dacă funcția de validare returnează `false`, formularul nu se va trimite la server

```
<form action="" method="post" onsubmit="return validareFormular()">
```

2. La metoda `onclick` a inputului de tip `button` al formularului. Funcția apelată va trebui să verifice inputurile și să îi ceară formularului să facă submit

```
<input type="button" onclick="return validareFormular()" value="Trimite">
```

OBSERVAȚII

JavaScript consideră pagina ca un array de controale. Astfel, `document.forms[0]` se referă la primul form de pe pagină.

```
document.forms[0].submit();
```

Pentru a seta o valoare unui control, folosim:

```
document.getElementById("idControl").value="valoare";
```