



## LABORATOR 6

<https://www.w3schools.com/>  
<https://developer.mozilla.org/>  
<https://www.w3.org/standards/webdesign/>

### DOM - Document Object Model – interacțiunea JS-ului cu elementele din pagina noastră web

DOM ([Document Object Model](#)) este o interfață de programare (API) a aplicațiilor care tratează un document HTML, XHTML sau XML ca o structură de arbore în care fiecare nod este un obiect reprezentând o parte a documentului. Obiectele pot fi manipulate în mod programat, cu ajutorul [funcțiilor DOM](#), iar eventualele schimbări vizibile aparute în urma executării lor pot fi reflectate în afișarea documentului.

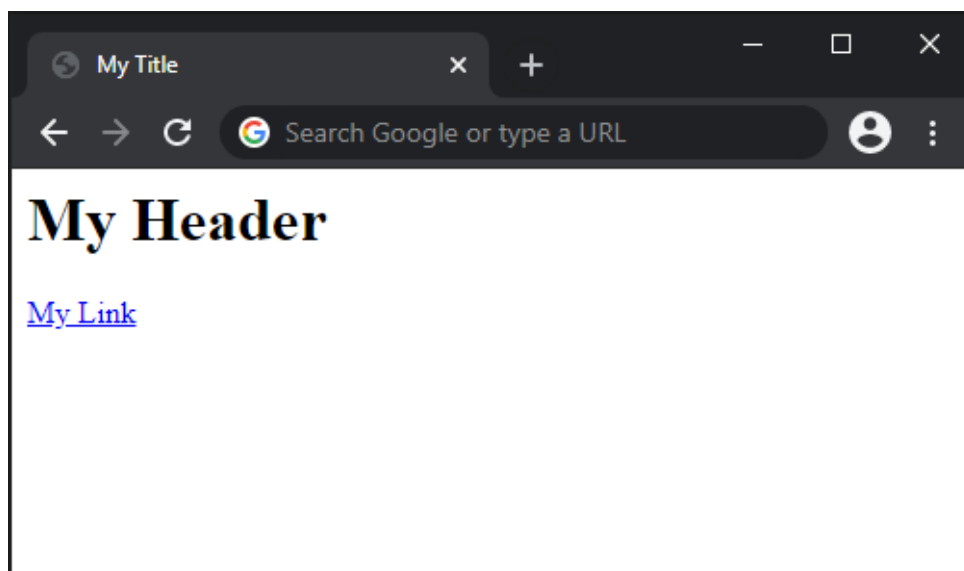
Deci, DOM-ul este un obiect creat de către browser la încărcarea paginii. DOM = arbore – nodurile = elementele HTML din pagina, de exemplu:

```
<html>

  <head>
    <title>My Title</title>
  </head>

  <body>
    <h1>My Header</h1>
    <a href="#" >My Link</a>
  </body>

</html>
```





Dupa ce pagina se incarca, DOM-ul este generat sub forma unui arbore, fix ca in imaginea de mai jos:

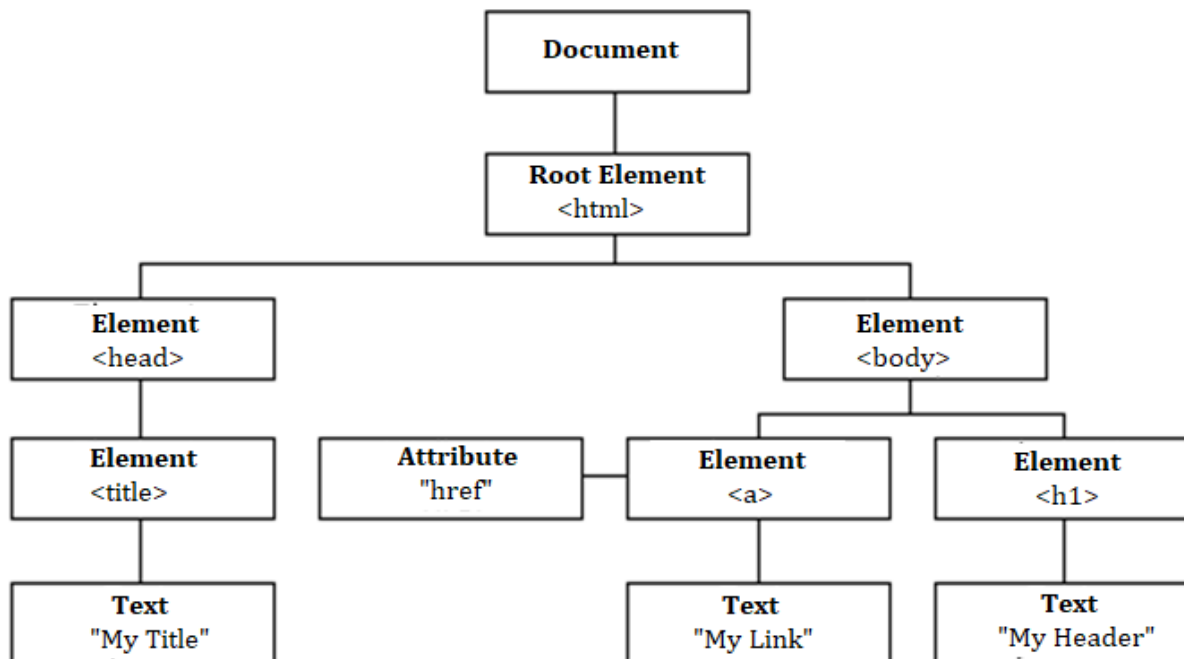


Image: Representation of the DOM tree.

Cum lucram cu DOM? – interaciunea cu elementele HTML inseamna selectarea acestora dupa **clasa**, **id** sau **tag** pentru adaugare de evenimente sau schimbare de proprietati/atribute.

- `getElementById` and `innerHTML`

This element has `id="blue_box"`. Can you get the text inside it and send it to the console? And can you change its content?

```
// selectarea unui element
```

```
document.getElementById("blue_box")
```

```
// selectarea continutului unui element
```

```
document.getElementById("blue_box").innerHTML;
```



UNIVERSITATEA DIN BUCUREȘTI  
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ



```
// schimbarea continutului selectat

    document.getElementById("blue_box").innerHTML = "This will be the
new content of the element.";

// trimiterea in console si schimbarea continutului selectat

    console.log(document.getElementById("blue_box").innerHTML );
    document.getElementById("blue_box").innerHTML = "Some new content";
```

## EXERCITII

- Sa se scrie o functie care primeste ca parametru un vector de numere intregi
- Creeaza (afiseaza, cu [document.write\(\)](#)) doua liste neordonate:
  - una cu titlul "numere pare" si cu elementele constand in numerele pare din vector si numarul lor
  - cealalta cu titlul "numere impare" si cu elementele constand in numerele impare din vector si numarul lor
- Folositi metode ale Array: push, length.

### EX.1 Hack Google

- Deschideti [google.com](http://google.com) in Chrome sau Firefox si afisati consola.
- Gasiti logo-ul Google si stocati-l intr-o variabila.
- Modificati sursa imaginii (url-ul) astfel incat sa apara logo-ul de la Yahoo. Puteti cauta url-ul logo-ului de la Yahoo intr-un motor de cautare.
- Gasiti butonul de search Google si stocati-l intr-o variabila.
- Modificati textul buttonului astfel incat sa scrie Yahoo.

### EX.2 - About Me Page

- Creati un document HTML (***aboutme.html***) cu template-ul de mai jos inclus in <body>:

```
<h1>About Me</h1>
<ul>
  <li>Nickname: <span id="nickname"></span></li>
  <li>Favorites: <span id="favorites"></span></li>
```



UNIVERSITATEA DIN BUCUREȘTI  
FACULTATEA DE MATEMATICĂ ȘI INFORMATICĂ



<li>Hometown: <span id="hometown"></span></li>

</ul>

- Creați un fișier JavaScript **script.js** și legați-l de documentul HTML folosind <script> la sfârșitul acestuia.
- Schimbați stilul din <body> astfel încât să avem font-family: *Arial, sans-serif*.
- Înlocuiți ficare <span> (*nickname, favorites, hometown*) cu informațiile voastre.
- Iterați prin fiecare <li> și adăugați clasa *list-item*.
- Creați un fișier CSS (conectați-l la document) cu stilurile necesare clasei *list-item* pentru a avea un text roșu.
- Creați un nou element <img> cu atributul **src** către o imagine cu voi. Adăugați acel element în pagina.

### EX.3 – IMDB

- Creați un document HTML (**filme.html**) cu titlul "Filmele mele favorite:", folosind tag-ul <h1>.
- Creați un fișier JavaScript **script.js** și legați-l de documentul HTML folosind <script> la sfârșitul acestuia.
- Creați un vector cu obiecte care să conțină informațiile filmelor favorite. Obiectul trebuie să includă *titlu* (string), *durata* (number), *actori* (array of strings) și o proprietate de tip boolean care să indice dacă filmul a fost vizualizat sau nu.
- Iterați prin vector și creați dinamic (folosind document.createElement) în pagina elemente <p> care să conțină titlul filmului.

### EX.4 – continuare IMDB

- Folosiți <ul> și <li> pentru a afișa filmele, alături de celelalte informații despre acestea.
- Adăugați în fiecare obiect din vector o proprietate reprezentând imaginea filmului și reflectați schimbarea în pagina.
- Adăugați proprietăți CSS care să diferentieze filmele vizualizate de cele nevizualizate din lista.