# 

**PROIECT PENTRU OBȚINEREA ATESTĂRII PROFESIONALE ÎN INFORMATICĂ**

**Profesor, Elev,  
Noață Ileana Popescu Constantin- Mădălin**

**Mai, 2021**

**PROIECT PENTRU OBȚINEREA ATESTĂRII PROFESIONALE ÎN INFORMATICĂ**

**SITE WEB, ROMÂNIA**

**Profesor, Elev,  
Noață Ileana Popescu Constantin- Mădălin**

**Mai, 2021**

**Cuprins**

[**Introducere** 3](#_Toc64141613)

[**Ce este un site web?** 3](#_Toc64141614)

[**Ce este o pagină web?** 3](#_Toc64141615)

[**Tipuri de site-uri web** 3](#_Toc64141616)

[**Ce limbaje de programare am folosit?** 4](#_Toc64141617)

[**• HTML** 4](#_Toc64141618)

[Noțiuni de bază 4](#_Toc64141619)

[Paragrafe 5](#_Toc64141620)

[Editarea textului 5](#_Toc64141621)

[Imagini 6](#_Toc64141622)

[Link-uri 6](#_Toc64141623)

[Liste 7](#_Toc64141624)

[HTML5 7](#_Toc64141625)

[**• CSS** 8](#_Toc64141626)

[Personalizare text 8](#_Toc64141627)

[Personalizare casetă/Box Model 9](#_Toc64141628)

[**• JavaScript** 11](#_Toc64141629)

[Noțiuni de bază 11](#_Toc64141630)

[Funcții 12](#_Toc64141631)

[Evenimente 13](#_Toc64141632)

[Structuri/Instrucțiuni 13](#_Toc64141633)

[**Cum am creat site-ul *România*?** 13](#_Toc64141634)

[**Pagina de start (index.html)** 14](#_Toc64141635)

[**Pagina cu informații (information.html)** 18](#_Toc64141636)

[**Pagina galerie (media.html)** 21](#_Toc64141637)

[**Pagina de final/despre (about.html)** 22](#_Toc64141638)

[**Bibliografie** 24](#_Toc64141639)

# **Introducere**

## **Ce este un site web?**

Noțiunea *site web* provine din expresia engleză web site și desemnează o grupă de pagini web multimedia (conținând texte, imagini fixe, animații ș.a.), accesibile în Internet în principiu orișicui, de obicei pe o temă anume, și care sunt conectate între ele prin așa-numite hiperlinkuri. Diversele situri web pot fi create de către o organizație, o persoană particulară, instituții publice etc. Inițial noțiunea apărea în limba română scrisă sub două forme: *sit web* și *site* (ca în limba engleză). Lingvistul George Pruteanu considera că varianta de preferat este *sait*.

## **Ce este o pagină web?**

Un sit web este alcătuit de regulă din mai multe pagini web. **O pagină web** este un document creat cu ajutorul limbajului de marcare HTML și (opțional) limbaje de programare cum ar fi PHP, ASP ș.a. fiind accesibil vizitatorilor prin intermediul protocolului HTTP, care transferă informația de la server la browser. Pagina web se numește așa deoarece, afișată pe un monitor, ea se aseamănă cu o pagină de ziar: de obicei paginile web au o lățime care încape în întregime pe ecran. În schimb, pagina poate fi chiar mult mai înaltă (adâncă) decât înălțimea ecranului, ea putând fi totuși ușor afișată cu ajutorul funcțiilor normale ale mausului și browserului folosite, prin "tragere" în sus și în jos. De asemenea, un sit web poate fi vizualizat pe orice dispozitiv conectat la Internet capabil să afișeze informații prin intermediul protocolului HTTP (unele telefoane mobile, PDA-uri, etc.).

Un sit alcătuit din mai multe pagini are de obicei o pagină inițială sau principală numită homepage, de la care pleacă legături către paginile interioare, secundare. Structurile și schemele de "navigare" din interiorul siturilor web sunt foarte diferite, în funcție de scopurile, dorințele și posibilitățile ofertantului de informații. De obicei această homepage este chiar pagina de start a sitului, pe care ofertantul de informații în web o face cunoscută la public drept punct de plecare pentru întregul sit web al său.

## **Tipuri de site-uri web**

Siturile web se pot clasifica după o mulțime de factori, dar principalul factor rămâne subiectul de activitate (sau conținutul) sitului. Din punct de vedere tehnologic un sit web poate fi alcătuit din orice tipuri de date și informații statice, camere de discuții, produse și servicii de vânzare, anunțuri, formulare de completat online, sunete digitalizate, clipuri video, imagini statice și animate, efecte speciale, meniuri dinamice și multe, multe altele. Vorbind la un nivel mai înalt, subiectul (tema) unui sit web poate fi: un așa-numit blog, portal web, catalog web, magazin virtual, bancă, universitate virtuală, bibliotecă, enciclopedie virtuală, revistă web, ziar web și aproape orice altceva.

În cazul site-ului meu web, el este un blog, site informativ în care se abordează informații despre România.

# **Ce limbaje de programare am folosit?**

**Limbajul de programare** este un limbaj formal de expresii și reguli (sau tehnici) valide de formulare a instrucțiunilor pentru un computer. Un limbaj de programare are definite un set de reguli sintactice și semantice. El dă posibilitatea programatorului să specifice în mod exact și amănunțit acțiunile pe care trebuie să le execute calculatorul, în ce ordine și cu ce date. Specificarea constă practic în întocmirea/scrierea programelor necesare ("programare").

**Limbajele și notiunile de bază folosite în site-ul România sunt:**

## **• HTML**

**HyperText Markup Language** este un limbaj de marcare utilizat pentru crearea paginilor web ce pot fi afișate într-un browser. Scopul HTML este mai degrabă prezentarea informațiilor – paragrafe, fonturi, tabele ș.a.m.d. – decât descrierea semanticii documentului.

### Noțiuni de bază

Deși au fost lansate diverse versiuni de-a lungul anilor, elementele de bază HTML rămân aceleași. Structura unui document HTML a fost comparată cu cea a unui sandwich. Deoarece un sandwich are două felii de pâine, documentul HTML are etichete HTML de deschidere și închidere. Aceste etichete, precum pâinea într-un sandviș, înconjoară orice altceva:

Imediat după eticheta HTML de deschidere, veți găsi capul documentului, care este identificat prin deschiderea și închiderea etichetei *head*. Capul unui fișier HTML conține toate elementele non-vizuale care ajută la funcționarea paginii.



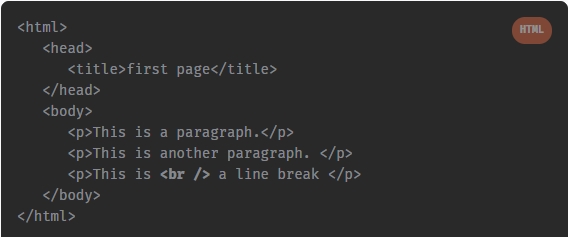
Eticheta *body* urmează eticheta *head*. Toate elementele vizual-structurale sunt conținute în eticheta *body*. Titluri, paragrafe, liste, citate, imagini și linkuri sunt doar câteva dintre elementele care pot fi conținute în eticheta corpului.

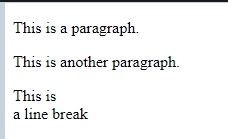


Eticheta <div> definește o diviziune sau o secțiune dintr-un document HTML.

### Paragrafe

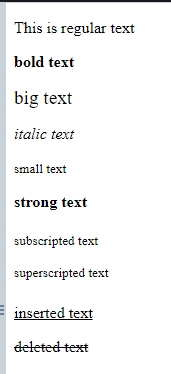
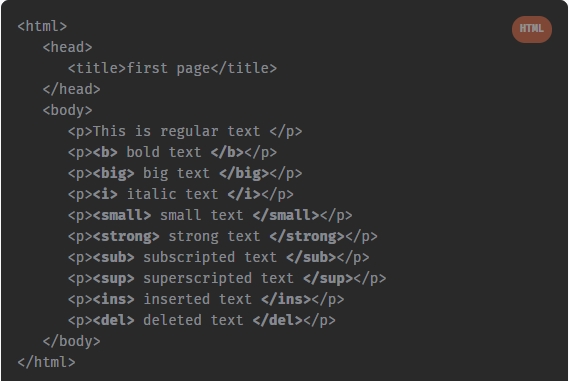
Pentru a crea un paragraf, pur și simplu se introduce elementul <p> cu etichetele de deschidere și închidere. Se folosește eticheta <br /> pentru a adăuga o singură linie de text fără a începe un nou paragraf.



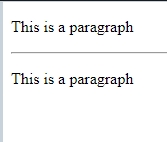
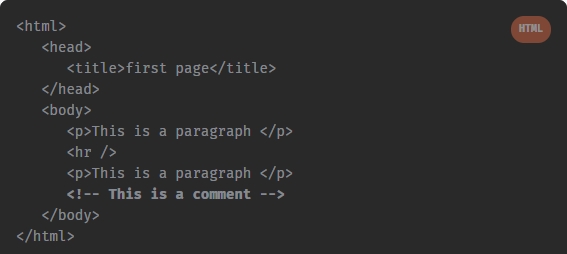


### Editarea textului

În HTML, există o listă de elemente care specifică stilul textului. Elementele de formatare au fost concepute pentru a afișa tipuri speciale de text:



HTML include șase niveluri de titluri, care sunt clasificate în funcție de importanță. Acestea sunt <h1>, <h2>, <h3>, <h4>, <h5> și <h6>. <hr /> - crează o linie orizontală.  
Browserul nu afișează comentarii, dar ajută la documentarea codului HTML și la adăugarea de descrieri și alte note.



### Imagini

Eticheta <img> este utilizată pentru a insera o imagine. Conține numai atribute și nu are o etichetă de închidere. Adresa URL a imaginii (adresa) poate fi definită folosind atributul *src*.

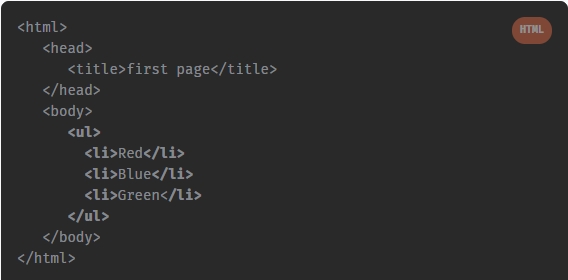
### Link-uri

Link-urile sunt, de asemenea, o parte integrantă a fiecărei pagini web. Puteți adăuga linkuri către text sau imagini care vor permite utilizatorului să facă click pe ele pentru a fi direcționat către un alt fișier sau pagină web. În HTML, linkurile sunt definite folosind eticheta <a>. Atributul *href* ajută pentru a defini adresa de destinație a link-ului.

Atributul *target* specifică unde se deschide documentul conectat. Cu valoarea *\_blank*, linkul va fi deschis într-o fereastră nouă sau o filă nouă.

### Liste

O listă începe cu eticheta <ol> și fiecare element de listă este definit de eticheta <li>. Se poate utiliza si eticheta <ul> pentru a pune puncte în loc de numere:



### HTML5

HTML5 este un limbaj pentru structurarea și prezentarea conținutului pentru World Wide Web (www), o tehnologie nucleu pentru Internet propusă inițial pentru software-ul Opera.

În documente HTML5, una dintre primele caracteristici noi este declarația de tip doc:



Declarația de codificare a caracterelor (charset) este, de asemenea, simplificată:



O structură de pagină HTML5 arată astfel:



## **• CSS**

**CSS sau Cascading Style Sheets** este un standard pentru formatarea elementelor unui document HTML. Stilurile se pot atașa elementelor HTML prin intermediul unor fișiere externe sau în cadrul documentului, prin elementul <style> și/sau atributul style. CSS se poate utiliza și pentru formatarea elementelor XHTML, XML și SVGL. CSS este unul dintre tehnologiile de bază utilizate în procesul de dezvoltare web, împreună cu HTML și JavaScript.

CSS permite separarea și prezentarea vizuală a conținutului unei pagini web, inclusiv culorile și fonturile disponibile. Separarea elementelor unei pagini îmbunătățește accesibilitatea paginii și permite o mai bună flexibilitate și un control în specificațiile caracteristicilor de prezentare.

### Personalizare text

Proprietatea *font-family* specifică fontul pentru un element. Există două tipuri de nume de familie de fonturi:

- familie de fonturi: o anumită familie de fonturi (cum ar fi Times New Roman sau Arial)   
- familie generică: un grup de familii de fonturi cu un aspect similar (cum ar fi Serif sau Monospace)

Proprietatea *font-size* setează dimensiunea unui font. O modalitate de a seta dimensiunea fonturilor pe web este utilizarea cuvintelor cheie. De exemplu xx-mic, mic, mediu, mare, mai mare etc. Sau se pot folosi valori urmate de unități de măsură precum px, em.

*Font-weight* controlează grosimea textului. Valorile pot fi setate ca normal (implicit), bold, bolder și lighter.

Proprietatea *color* din CSS specifică culoarea textului. O metodă de a specifica culoarea textului este utilizarea unui nume de culoare: cum ar fi roșu, verde, albastru etc. Un alt mod de definire a culorilor este utilizarea valorilor hexazecimale și RGB. Forma hexazecimală este un semn (#) urmat de cel mult 6 valori hex (0-F). RGB definește valorile individuale pentru roșu, verde și albastru.

Proprietatea *text-align* specifică alinierea orizontală a textului într-un element. În mod implicit, textul este aliniat la stânga.

Proprietatea *text-decoration* specifică modul în care va fi decorat textul.Valorile utilizate în mod obișnuit sunt:

- none - Valoarea implicită, aceasta definește un text normal

- inherit - Moștenește această proprietate de elementul său părinte

- overline - Desenează o linie orizontală deasupra textului

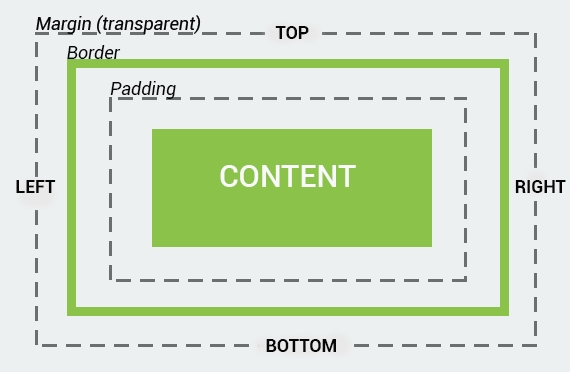
- underline - Desenează o linie orizontală sub text

- line-through - trasează o linie orizontală prin text (înlocuiește eticheta HTML <s>)

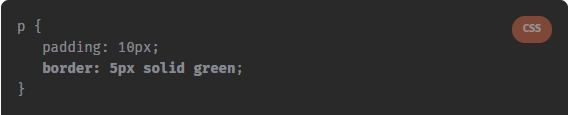
Proprietatea *text-shadow* adaugă umbră textului. Este nevoie de patru valori: prima valoare definește distanța umbrei în direcția x (orizontală), a doua valoare stabilește distanța în direcția y (verticală), a treia valoare definește neclaritatea umbrei și a patra valoare setează culoarea.

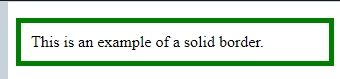
### Personalizare casetă/Box Model

Toate elementele HTML pot fi considerate ca niște casete. Modelul de casetă CSS reprezintă designul și aspectul site-ului. Se compune din margins, borders, paddings și conținutul real. Proprietățile funcționează în aceeași ordine: sus, dreapta, jos și stânga. Imaginea de mai jos ilustrează modelul cutiei:



Proprietatea *border* permite personalizarea marginile elementelor HTML. Pentru a adăuga un chenar la element, trebuie specificată dimensiunea, stilul și culoarea chenarului. Exemplul de mai jos arată cum se formează o margine verde solidă la paragraf.



Valoarea implicită a *border-style* este none, care nu definește nici o margine. Există diferite stiluri acceptate pentru proprietatea *border-style*: dotted, dashed, double, etc.

Proprietatea *background-color* este utilizată pentru a specifica culoarea de fundal a unui element.

Proprietatea *background-image* setează una sau mai multe imagini de fundal într-un element.  
*url* specifică calea către fișierul imagine. Sunt acceptate atât căile relative, cât și cele absolute.  
Proprietatea *background-repeat* specifică modul în care se repetă imaginile de fundal.

Linkurile pot fi stilizate cu orice proprietate CSS (de exemplu color, font-family, etc.).  
În plus, linkurile pot fi stilate diferit, în funcție de starea în care sunt. Următorii pseudo-selectori sunt disponibili:

a: link - definește stilul pentru linkurile nevizitate normale

a: visited - definește stilul pentru linkurile vizitate

a: active - un link devine activ odată ce faceți clic pe el

a: hover - se leagă un link când mouse-ul este deasupra acestuia

Proprietatea *display* specifică dacă / cum este afișat un element. Fiecare element HTML are o valoare de afișare implicită, în funcție de ce tip de element este. Valoarea implicită de afișare pentru majoritatea elementelor este *block* sau *inline*.

Proprietatea *opacity* specifică opacitatea / transparența unui element.

Gradientele CSS permit afișearea tranzițiilor fluide între două sau mai multe culori specificate. CSS definește două tipuri de gradiente: - Gradienți lineari (merg în jos / sus / stânga / dreapta / în diagonală)

- Gradienți radiali (pornesc din centru)

## **• JavaScript**

**JavaScript (JS)** este un limbaj de programare orientat pe obiecte bazat pe conceptul prototipurilor.[5] Este folosit mai ales pentru introducerea unor funcționalități în paginile web, codul JavaScript din aceste pagini fiind rulat de către browser. Limbajul este binecunoscut pentru folosirea sa în construirea siturilor web, dar este folosit și pentru accesul la obiecte încapsulate (embedded objects) în alte aplicații.

### Noțiuni de bază

În HTML, codul JavaScript este inserat între etichetele <script> și </script>.

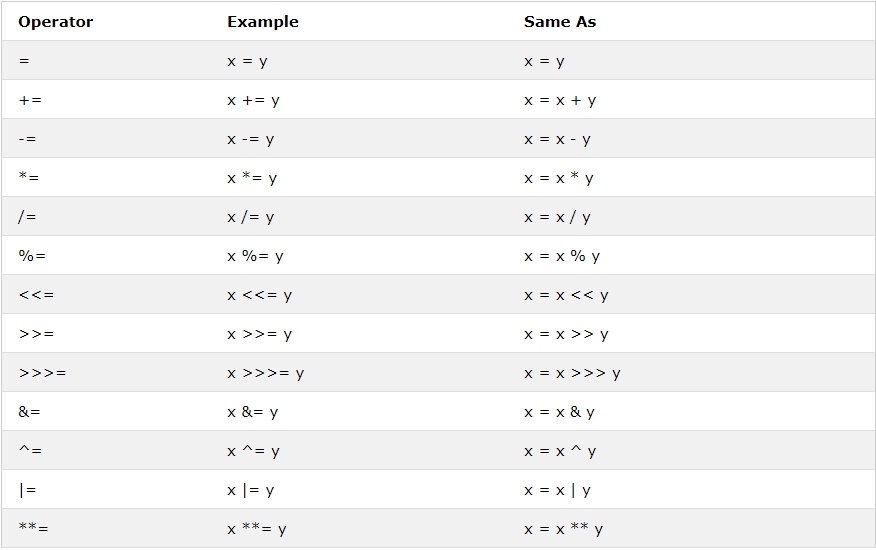
JavaScript poate „afișa” datele în diferite moduri:

* Scrierea într-un element HTML, folosind innerHTML.
* Scrierea în ieșirea HTML folosind document.write ().
* Scrierea într-o casetă de alertă, utilizând window.alert ().
* Scrierea în consola browserului, folosind console.log ().

Pentru a accesa un element HTML, JavaScript poate utiliza metoda document.getElementById (id). Atributul id definește elementul HTML.

Într-un limbaj de programare, variabilele sunt utilizate pentru a stoca valorile datelor. JavaScript folosește cuvântul cheie *var* pentru a declara variabile. Un semn egal este utilizat pentru a atribui valori variabilelor. JavaScript folosește operatori aritmetici (+ - \* /) pentru a calcula valorile. Codul după bare oblice // sau între / \* și \* / este tratat ca un comentariu. Comentariile sunt ignorate și nu vor fi executate.

Operatorii de atribuire atribuie valori variabilelor JavaScript:



### Funcții

O funcție JavaScript este un bloc de cod conceput pentru a efectua o anumită sarcină. O funcție JavaScript este executată atunci când „ceva” o invocă (o numește).

O funcție JavaScript este definită cu cuvântul cheie *function*, urmat de un nume, urmat de paranteze (). Numele funcțiilor pot conține litere, cifre, puncte de subliniere și semne de dolar (aceleași reguli ca variabilele). Parantezele pot include nume de parametri separați prin virgule: (parametru1, parametru2, ...). Codul care trebuie executat, prin funcție, este plasat în paranteze: {}

Când JavaScript ajunge la o declarație return, funcția se va opri din executare. Dacă funcția a fost invocată dintr-o instrucțiune, JavaScript va „reveni” pentru a executa codul după instrucțiunea de invocare. Funcțiile calculează adesea o valoare returnată. Valoarea returnată este „returnată” înapoi la „apelant”.

### Evenimente

Evenimentele HTML sunt „lucruri” care se întâmplă elementelor HTML. Când JavaScript este utilizat în paginile HTML, JavaScript poate „reacționa” la aceste evenimente.

O listă cu câteva evenimente HTML obișnuite:

* onchange - Un element HTML a fost modificat
* onclick - Utilizatorul face clic pe un element HTML
* onmouseover - Utilizatorul deplasează mouse-ul peste un element HTML
* onmouseout - Utilizatorul îndepărtează mouse-ul de un element HTML
* onkeydown - Utilizatorul apasă o tastă de tastatură
* onload - Browserul a terminat încărcarea paginii

JavaScript poate modifica ,,setările,, CSS ale elementelor HTML cu ajutorul sitagmei *.style* care se atribuie variabilei descrise de id-ul elementului HTML.

### Structuri/Instrucțiuni

Instrucțiunea *if* se utilizează pentru a specifica un bloc de cod JavaScript care trebuie executat dacă o condiție este adevărată. *Else* se folosește dacă condiția din if este falsă. Se poate folosi *else if* dacă există o nouă condiție.

JavaScript acceptă diferite tipuri de bucle:

- for - parcurge de mai multe ori un bloc de cod

- for / in - parcurge proprietățile unui obiect

- for / of - parcurge valorile unui obiect iterabil

- while - parcurge un bloc de cod în timp ce o condiție specificată este adevărată

- do / while – parcurge, de asemenea, un bloc de cod în timp ce o condiție specificată este adevărată

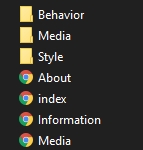
Declarația *break* „sare” dintr-o buclă. Instrucțiunea *continue* „sare peste” o iterație în buclă.

# **Cum am creat site-ul *România*?**

Site-ul prezentat are ca temă țara România și așa cum am specificat mai sus, am folosit JavaScript, HTML și CSS pentru a-l crea.

Am folosit ca și IDE, Visual Code. Un mediu de dezvoltare (IDE) este un set de programe care ajută programatorul în scrierea programelor. Un mediu de dezvoltare combină toți pașii necesari creării unui program într-un singur soft, care, de regulă, oferă o interfață cu utilizatorul grafică, prietenoasă.

**Structura site-ului**



Codul scris în HTML este pus în folderul principal astfel încât să se poată accesa celelalte elemente fără a fi nevoie să se specifice adresa exactă a fișierului utilizat pentru a nu se crea erori în momentul în care folderul cu tot conținutul este mutat. Fișierul *index.html* conține pagina prinpală care are legătură cu fișierul *Information.html*, iar apoi fișierele *Media.html*, *About.html* și *Information.html* sunt legate între ele. Folderul *Media* conține imaginile site-ului, fără imaginile de fundal deoarece acestea sunt poziționate în folderul *Style* împreună cu liniile din CSS, care oferă un alt aspect elementelor HTML, pentru a nu se utiliza întreaga adresă a poziționări imaginilor în calculator. Folderul *Behavior* conține codurile JavaScript care ajută site-ul la anumite animații și crează anumite comportamente obiectelor.

## **Pagina de start (index.html)**

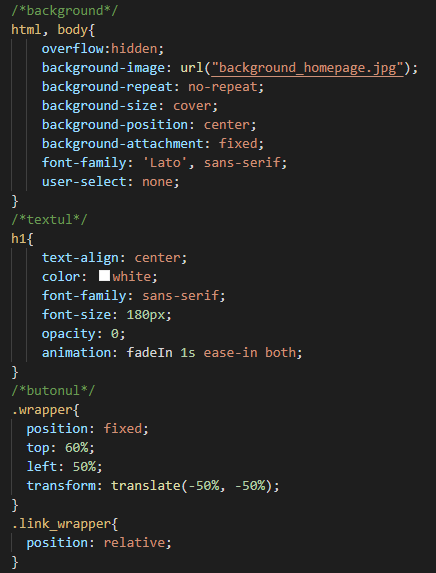


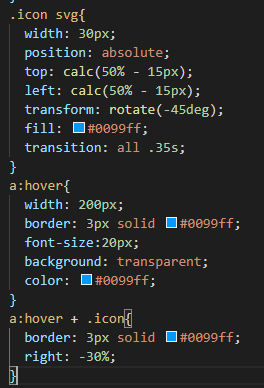
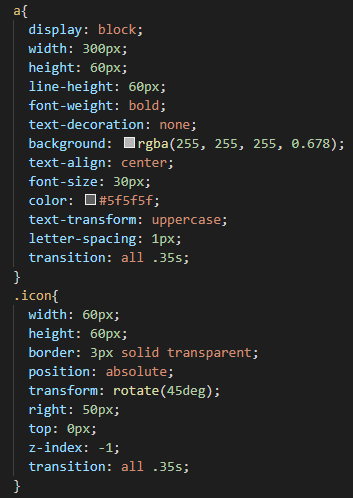
Conține un titlu și un link care este personalizat ca și un buton care duce la pagina de informații (information.html).

Cât despre cod, în toate fișierele HTML se regăsește începutul de HTML5, care ușurează munca, titlul împreună cu iconița de la tab și legătura cu fișierele CSS.

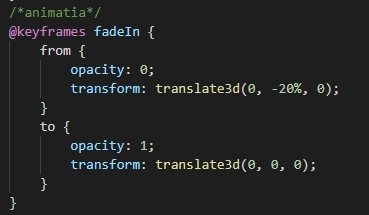


Imaginea de fundal am asimilat-o tagului *body*, *html* unde am blocat posibilitatea de scroll și de a selecta text. Pentru titlu (*h1*) și buton, pe lângă aspectul creat, am adăugat și opacitatea 0 dar și faptul ca urmează o animație. Butonul este creat din mai multe blocuri (dreptunghi, text, romb, săgeată) care inițial sunt transparente, iar la stadiul de *hover* se redimensionează și au culoarea albastru.

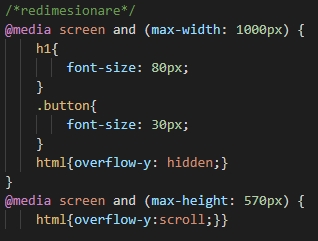




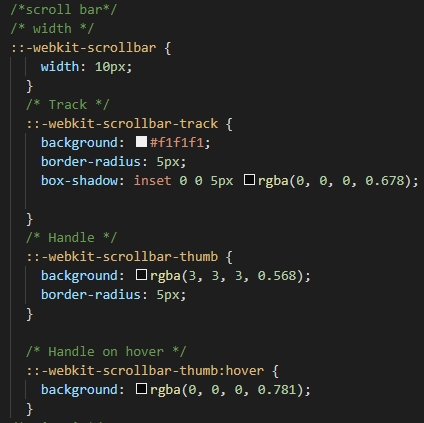
Animația crește opacitatea la 1 și duce obiectele de sus în poziția inițială.



Cum am oprit funcția de scroll, a trebuit să adaug o funcție care să îmi redimensioneze obiectele pentru a nu se vedea doar o parte din ele pe fereastra redimensionată.

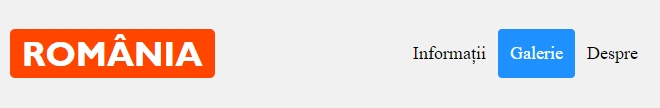


Bara de scroll este personalizată, deși în această pagină apare doar când fereastră este micșorată:



## **Pagina cu informații (information.html)**

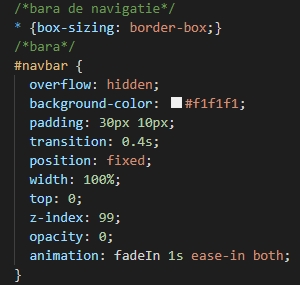
Toate celelalte pagini conțin o bară de navigatie, unde se pot accesa între ele, dar se poate accesa și pagina principală/de start (index.html).



Butoanele sunt niște link-uri personalizate, care sunt puse într-un *div* iar ultimele 3 sunt puse în altul pentru a le poziționa la dreapta.



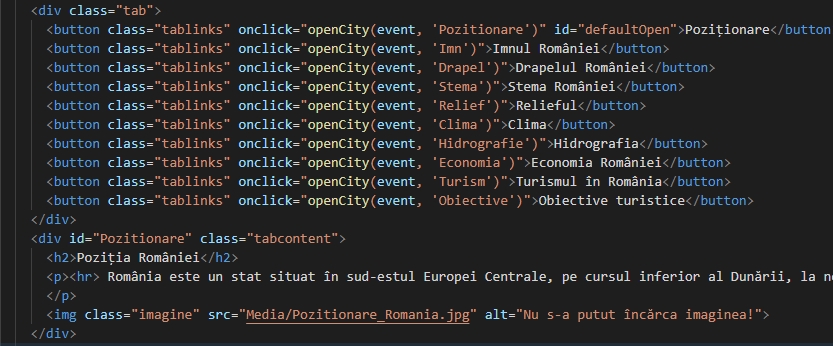
Bara este o casetă care la fel are o animație de apariție, dar și una făcută în JS, care în momentul în care se dă scroll, bara se redimensionează.



Redimensionarea apare și pe această pagină.

În plus, această pagină mai are un meniu, în care sunt expuse titlurile informațiilor geografice despre România și pentru fiecare apare informația care definește titlul.

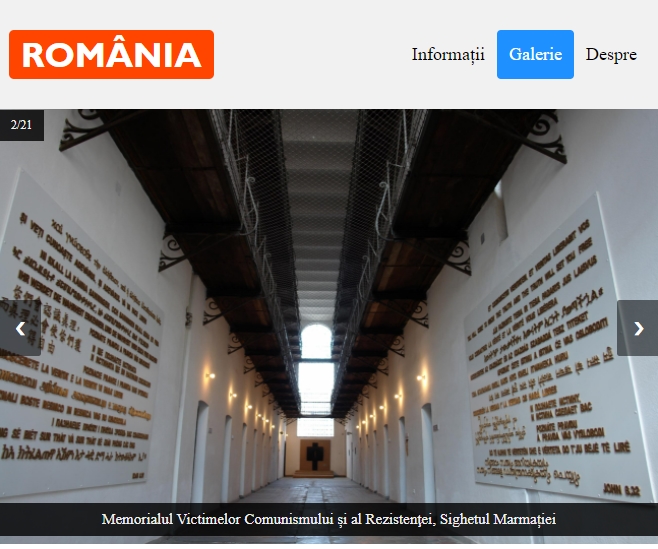




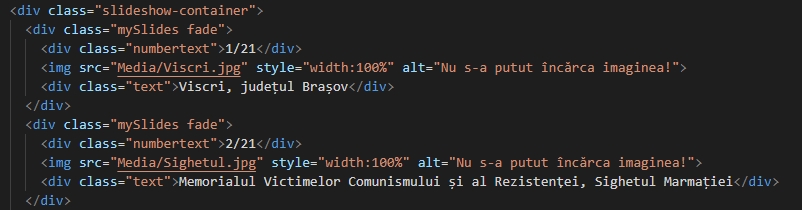
La apăsarea unui buton se apelează funcția openCity din JS și ea scoate din afișare contentul celorlalte butoane și îl afișează pe cel selectat.



## **Pagina galerie (media.html)**



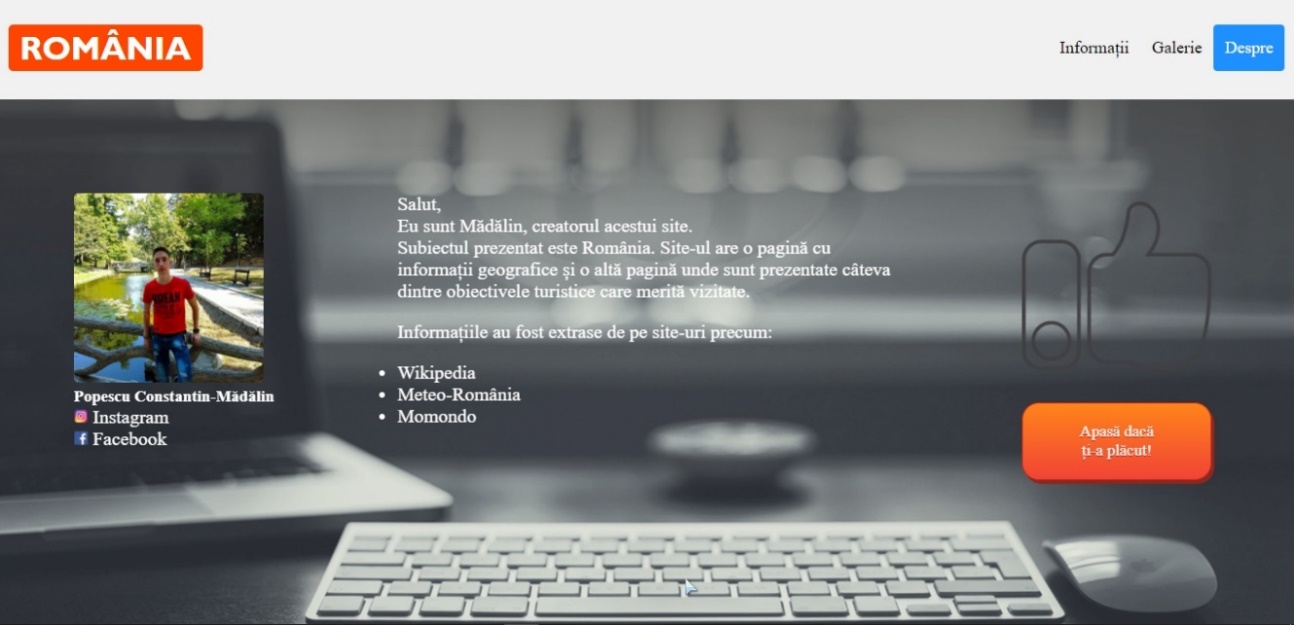
În această pagină am preferat să deblochez funcția scroll, să setez lățimea imaginilor cât ecranul și înălțimea să fie în funcție de mărimea imagini, astfel scroll-ul să fie doar vertical. Am considerat că este o perspectivă mai bună de vizionat imaginile, decât să fie micuțe și să se piardă din calitate mai mult decât trebuit. În stânga sus se află numârul lor, jos se află numele zonei, iar în parți, la mijlocul înălțimii imaginilor se află sâgețile de înainte și înapoi.



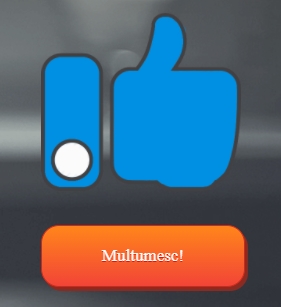
Codul constă într-un *div* mare care conține un *div* pentru fiecare imagine. Funcționalitatea este făcută în JavaScript, iar partea de personalizare în CSS.



## **Pagina de final/despre (about.html)**



Aici se află informații referitoare la mine și la informațiile găsite pe site. Este alcătuită, pe lângă bara de navigație, de un *div* unde se află link-uri referitoare la rețele de socializare, unde pot fi contactat, un alt *div* unde spun câteva cuvinte și am creat o listă cu link-urile de unde am luat în mare parte informațiile și ultimul *div* în care este ceva interativ unde dacă se apasă butonul, like-ul prinde culoare și butonul își schimbă textul.



Există și o funcționalitate în JS, pentru a face ca în momentul în care butonul se apasă din nou, culoarea imagini să dispară și textul să revină.



# **Bibliografie**

* ro.wikipedia.org
* sololearn.com
* w3schools.com