

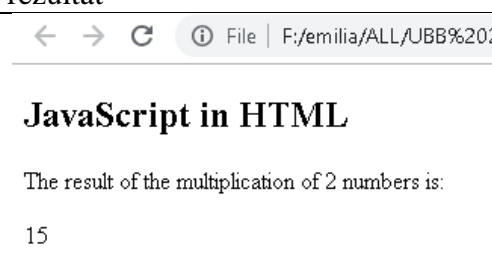
Exercitii pentru student – JAVASCRIPT

Exemplu 1:

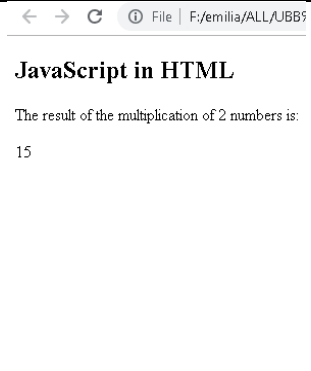
index.html	rezultat
<pre><!DOCTYPE html> <html> <body> <h1>JavaScript test</h1> <button type="button" onclick="document.getElementById('test').innerHTML = Date()"> Click to display Date and Time.</button> <p id="test"></p> </body> </html></pre>	

Exemplu 2:

Varianta 1: tot codul va fi fisier .html

index.html	rezultat
<pre><!DOCTYPE html> <html> <body> <h2>JavaScript in HTML</h2> <p> The result of the multiplication of 2 numbers is:</p> <p id="demo"></p> <script> var x = myFunction(5, 3); document.getElementById("demo").innerHTML = x; function myFunction(a, b) { return a * b; } </script> </body> </html></pre>	

Varianta 2: codul va fi separat in fisiere: .html, .js, .css (daca este cazul)

index.html	page.js	rezultat
<pre><!DOCTYPE html> <html> <body> <h2>JavaScript in HTML</h2> <p>The result of the multiplication of 2 numbers is:</p> <p id="test"></p> <script src="page.js"></script> </body> </html></pre>	<pre>var x = myFunction(5, 3); document.getElementById("test").innerHTML = x; function myFunction(a, b) { return a * b; }</pre>	

Indicatii la probleme:

1. Sa se scrie o pagina HTML ce contine doua liste cu mai multe elemente fiecare, create cu ajutorul tagului `<select>`. La un dublu click pe un element al primei liste, acesta va fi mutat in lista a doua si invers. Nu se vor folosi biblioteci de functii, jQuery, pluginuri, etc.

Se pot folosi 2 `<select>`-uri

```
<select size="10" id="list1">
```

```
<option id="o1" ondblclick="function_name(this.id)" value="value opt 1">opt 1</option>
```

unde, `function_name(this.id)` va fi o functie javascript

... cate optiuni se doresc ...

```
</select>
```

Apelul functiei javascript va fi de forma `<script src="program.js"></script>`, unde `program.js` contine functia javascript `function_name(parameter)`.

In functia javascript se poate folosi metoda `document.getElementById(el)`.

De exemplu:



- Un formular web va permite unui utilizator sa isi introduca numele, data nasterii, varsta si adresa de e-mail. La apasarea unui buton “Trimite” se vor valida toate aceste campuri daca sunt completate si daca sunt completate corect. Daca da, se va afisa un mesaj “Datele sunt completate corect”, altfel, se va afisa un mesaj de genul “Campurile nume si varsta nu sunt completate corect”, aceste campuri fiind “incercuite” intr-o bordura rosie. Toate aceste validari vor fi implementate pe client in JavaScript. Nu se vor folosi biblioteci de functii, jQuery, pluginuri, etc.

Se poate lucra cu un div ce contine cele 4 input-uri: Nume, Data Nastere, Varsta si Adresa de e-mail, avand tipurile corespunzatoare (de exemplu, tipul date pentru Data Nastere).

Pentru fiecare input se vor face functii de validare (sa nu fie vide si corespunzator):

- Nume – sa contina doar litere
- Varsta - fie cuprinsa intre 1 si 99, fie sa fie chiar varsta persoanei, calculata ca si diferenta dintre data actuala si Data Nasterii
- Adresa de e-mail – sa contina caracterul @ si .-ul (punctual) corespunzator domeniului adresei de e-mail; sa contina caractere inainte de @ si inainte si dupa .

The screenshot shows a web form with four input fields: Name (containing 'ana'), Birth (containing '04/04/2019'), Age (containing '12'), and Email (containing 'a@gmail.com'). Each field has a green border, indicating it is valid. A 'TRIMITE' button is at the bottom left. To the right, a message box says 'This page says: Everything alright' with an 'OK' button.

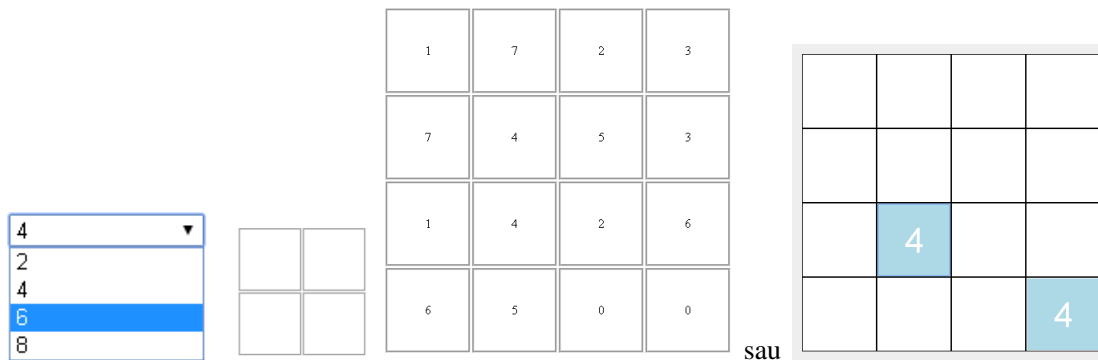
The screenshot shows the same web form, but the Name field now contains 'a2' and has a red border. The Birth, Age, and Email fields remain valid with green borders. The 'TRIMITE' button is at the bottom left. To the right, a message box says 'This page says: Invalid name' with an 'OK' button.

The screenshot shows the web form with multiple invalid inputs: Name ('ana23'), Age ('455'), and Email ('om') all have red borders. The Birth field ('04/04/2019') remains valid with a green border. The 'TRIMITE' button is at the bottom left. To the right, a message box says 'This page says: Invalid name, Invalid age, Invalid email' with an 'OK' button.

- Sa se implementeze folosind JavaScript urmatoarea problema. O matrice cu numar par de elemente, reprezentata vizual sub forma unui tabel, contine perechi de numere initial ascunse. Daca utilizatorul da click pe doua celule ale tabelului ce contin numere egale acestea vor fi afisate si vor ramane afisate. Daca numerele continute in cele doua celule nu sunt egale, vor fi ascunse din nou dupa un numar de 2, 3 secunde. Jocul se termina cand toate perechile de numere au fost ghicite. Dupa prima implementare a jocului, se va crea o noua versiune in care numerele vor fi inlocuite cu imagini (ce contin fructe spre

exemplu, sau “profi” de pe pagina facultatii). Problema mai este recunoscuta si sub numele de Memory Game. Nu se vor folosi biblioteci de functii, jQuery, pluginuri, etc.

Problema se poate realiza pentru 2, 4, 6, 8, ... celule (care se vor selecta la inceputul jocului) sau pentru un numar fix de celule (doar 4 de exemplu, de unde rezulta o matrice de 4x4).



Problema va avea 2 versiuni: una cu numere si cealalta cu poze.



4. Sa se scrie o pagina HTML ce contine un tabel cu cel putin trei coloane si cel putin trei linii. Prima coloana reprezinta antetul. Cand utilizatorul va da click pe o celula din antet, elementele din tabel se vor sorta crescator sau descrescator in functie de valorile prezente pe linia corespunzatoare antetului pe care s-a dat click. Codul JavaScript va fi reutilizabil si va permite crearea unui comportament de tabel sortabil pentru mai multe tabele existente in cadrul paginii. Nu se vor folosi biblioteci de functii, jQuery, pluginuri, etc. Exemplu de tabel ce se doreste a fi sortat (sortarea se va face alfabetic dupa numele fructului, dupa pret sau dupa cantitate):

Fructe	Mere	Pere
Pret	3	4
Cantitate	8	6

Dupa rezolvarea problemei, implementati o noua varianta in care capul de tabel este orizontal, nu vertical. Nu se vor folosi biblioteci de functii, jQuery, pluginuri, etc.

Se vor rezolva 2 probleme: una cu ordonare pe coloana (tabelul se va crea adaugand inregistrările corespunzatoare fiecarui rand – Mere 3 8) si cealalta cu ordonare pe linie (tabelul se va crea pe coloana – Mere Pere Prune Struguri Nuci). Se vor aplica functii de ordonare.

Fructe	Pret	Cantitate
Mere	3	8
Nuci	50	20
Pere	4	6
Prune	4	10
Struguri	10	40

Produs	Mere	Nuci	Pere	Prune	Struguri
Pret	3	50	4	4	10
Cantitate	8	20	6	10	40

Fructe	Pret	Cantitate
Mere	3	8
Pere	4	6
Prune	4	10
Struguri	10	40
Nuci	50	20

Produs	Mere	Pere	Prune	Struguri	Nuci
Pret	3	4	4	10	50
Cantitate	8	6	10	40	20

Fructe	Pret	Cantitate
Mere	3	8
Pere	4	6
Prune	4	10
Struguri	10	40
Nuci	50	20

Produs	Pere	Mere	Prune	Nuci	Struguri
Pret	4	3	4	50	10
Cantitate	6	8	10	20	40

5. Intr-o pagina HTML exista o lista descrisa cu ajutorul tagului ``. Fiecare element din lista (``) contine o imagine (``), si un link (`<a>`). Elemente listei, mai putin primul dintre ele, nu sunt vizibile initial (se poate folosi in acest sens CSS). Afisarea unui element din lista presupune afisarea imaginii si a textului ca link peste imagine (a se vedea ca exemplu carouselul din pagina facultatii). Dupa n secunde printr-un efect de tranzitie va fi afisat urmatorul element din lista. Se vor implementa si doua butoane Next si Previous care vor permite afisarea elementelor urmatoare sau anterioare fara a se astepta trecerea celor n secunde. Nu se vor folosi biblioteci de functii, jQuery, pluginuri, etc.

Se poate realiza un filmulet de imagini cu Facebook, google, gmail, yahoo, ... ☺ ce contine sageti pentru a trimite de la o imagine la alta si/sau merge si automatic din x secunde in x secunde. Cand se va da click pe o imagine se va redirecta spre un link (preferabil cel indicat de imagine).



Se va crea o functie ce va permite accesarea imaginilor anterioare si urmatoare si o functie pentru slide show pentru imagini. Imaginele se pot pune intr-o lista ordonata si hyperlink catre pagina pe care o reprezinta.

- O pagina HTML va contine un tabel cu n linii si n coloane. Celulele tabelului contin numere de la 1 la n^2 intr-o ordine aleatoare. Una dintre celule este libera. Folosind JavaScript sa se creeze in cadrul paginii un joc de tip puzzle care la apasarea tastelor sageti va interschimba celula libera cu celula adiacenta (corespunzatoare tastei sus, jos, stanga, dreapta apasata). Nu se vor folosi biblioteci de functii, jQuery, pluginuri, etc.

Puzzle-ul initial:					Puzzle-ul rezolvat:				
	3	8	9	5		1	2	3	4
	7	13	6	15		5	6	7	8
	10		14	4		9	10	11	12
	2	11	1	12		13	14	15	

Cerinta facultativa pentru cei ce se “plictisesc”: Dupa implementarea initiala, se va crea o a doua varianta a jocului, unde, in loc de numere, celulele tabelului vor contine sprite-uri CSS obtinute din cadrul unei aceleasi imagini patrate. Numarul n se va da la inceputul jocului, la fel si URL-ul imaginii ce se va folosi in sprite-urile CSS ca si fundal pe celulele tabelului. Nu se vor folosi biblioteci de functii, jQuery, pluginuri, etc.

Se pot considera 2 cazuri: unul in care se genereaza puzzle-ul cu o dimensiune data sau conform unui numar introdus la inceput se poate genera un puzzle de dimensiuni diferite.

