

Examen Tehnici Web seria 13 – 17.06.2021

Regulament:

- fiecare subiect se va rezolva într-un fișier separat cu extensia html (ex: p1.html, p2.html, p3.html, p4.html), codurile CSS și JavaScript corespunzătoare fiecărui exercițiu vor fi scrise direct în fișierul html, excepție făcând fișierul separat JavaScript (notat de obicei **app.js**) necesar pentru rezolvarea subiectului de Node.js.
- rezolvarile se trimit sub forma unei arhive (.zip), a carei nume este de forma **grupa_nume_prenume_varianta**, care trebuie să includă și fișierul cu subiectul primit prin email; arhiva este trimisă prin email, răspunzând la mesajul prin care ați primit subiectele **SI** se încarcă folosind următorul link: <https://tinyurl.com/ab442x2p>
- fiecare exercițiu are **1,5 puncte**.
- condiția de promovare a testului: **minim 3 puncte**.
- **Important!** Orice similaritate a codului se pedepsește cu anularea punctajului subiectului/subiectelor pentru toți cei implicați.

Recomandări:

- dacă nu știți să creați dinamic (prin JavaScript) elementele cerute, pentru a putea finaliza exercițiul, se pot adăuga static elementele (cu scăderea punctajului alocat creării dinamice).
- dacă nu știți să lucrați cu colecții, realizați cerința macar pentru un singur element al colecției.
- nu tratați alte cazuri decât cele cerute explicit de problema.

Tel. situații urgente: 0768945065 (Carmen Chirita)

P1. Creați un element div cu id-ul "parinte" și care să aibă în interiorul său (copii directi) alte 3 divuri cu clasa "patrat". Scrieți cod CSS care să rezolve cerințele:

- la lățimea paginii peste 800px, divul cu id-ul "parinte" să fie centrat (pe orizontală) în pagină, să aibă înălțimea și lățimea de 600px, background verde și border negru de 1px iar divurile cu clasa "patrat" de pe pozițiile pare să aibă border roșu iar cele de pe poziții impare border albastru și să fie poziționate în interiorul divului parinte pe 3 coloane de lățimi egale și având înălțimi egale cu ale divului parinte.
- la lățimea paginii sub 800px divul cu id-ul "parinte" va avea lățimea egală cu jumătate din lățimea paginii iar elementele din clasa "patrat" se vor dispune pe o singură coloană, unul sub altul, ocupând întreaga lățime a părintelui.
- tot la lățimea paginii sub 800px, culoarea de background a paginii se va schimba lent timp de o secundă, de la alb la roz și invers repetându-se la infinit.

P2. Creați un document HTML fără conținut (nu conține alte elemente în interiorul body-ului).

Scrieți **cod JavaScript** pentru a rezolva următoarele:

- la încărcarea paginii se vor crea dinamic N divuri, unde N este ora la care a fost deschisă pagină, fiecare div primind la creare clasa "cutie". Scrieți cod CSS astfel încât clasa "cutie" să atribuie o lățime și înălțime egale cu 10px și border negru de 2px. În plus, fiecare div va avea o culoare de background generată aleator dintr-un vector de culori alese de voi.
- fiecare click pe un div, va face să crească dimensiunile acestuia cu 5px. La click pe body dar în afara divurilor, toate divurile revin la dimensiunile inițiale. Click-ul pe un div nu ar trebui să declanșeze click-ul pe body.

P3. Presupunem ca intr-un document HTML avem un grup de butoane radio cu etichetele "10px", "20px", "30px", "40px", "50px" cu proprietatea ca nu pot fi 2 butoane selectate in acelasi timp. Scrieti cod JavaScript pentru a rezolva:

- la incarcarea paginii se va afisa intr-un paragraf din pagina numele vostru si data la care a fost deschisa pagina si la fiecare 3s cate un buton radio se va selecta automat iar paragraful va primi un border solid de grosime egala cu valoarea etichetei butonului radio selectat. Dupa ultimul buton, selectia se reia de la inceput (de la primul).
- daca se apasa o tasta cifra (0-9) selectia se opreste iar la o noua reincarcare a paginii va fi selectat automat butonul la care s-a oprit selectia inainte de reincarcare.

P4. Scrieti un formular cu un element select avand optiunile "aventura", "fantastic" si "politist" si un buton-submit. Se considera pe server vectorul de obiecte,
carti = [{titlu:"c1", gen:"aventura", pagini:"200"}, {titlu:"c4", gen:"politist", pagini:"240"}, {titlu:"c5", gen:"fantastic", pagini:"300"}, {titlu:"c2", gen:"fantastic", pagini:"450"}, {titlu:"c3", gen:"aventura", pagini:"600"}].

Scrieti aplicatia server **app.js** astfel incat dupa selectarea unei optiuni, la click pe butonul de submit, optiunea selectata se va trimite catre server iar in pagina se va afisa (in locul formularului sau sub el) toate cartile (titlurile) care apartin genului selectat si care au mai mult de 250 de pagini sau mesajul **"Nu exista carti"** daca nu exista nici o carte care indeplineste conditiile mentionate.