

Informații laborator și proiect

Punctajul maxim care poate fi obținut în cadrul laboratorului este de 40 puncte: 10 puncte activitate laborator și 30 puncte proiect. Punctajul minim de intrare în examen este 20 puncte.

Activitate laborator(maxim 10 puncte)

Punctele pentru activitatea din cadrul laboratorului se obțin astfel:

- Minim 10 prezențe(maxim 5 puncte)
- Toate laboratoarele rezolvate(maxim 5 puncte)

Proiect(maxim 30 puncte)

Cerințe HTML+CSS (10 puncte)

- fișiere separate pentru HTML(cel puțin 3 pagini diferite) și CSS;
- HTML: proiectul sa folosească tagurile semantice prezentate la curs/laborator;
- sa treacă testul de validare HTML (<http://validator.w3.org/>) și testul de validare CSS (<https://jigsaw.w3.org/css-validator/>);
- site-ul sa fie responsive (media query, unitati relative, tipuri de display);
- selectori CSS de bază (după id, clasă, tag, elementContinut, elementCopil, al n-lea copil de tipul unui tag (:nth-of-type));
- proprietăți: width, height, color, background, font-size, border, padding, margin, display;
- layout-ul să conțină coloane (pe minim o pagină) și fie realizat fără tabele (folosind flex și grid);
- să conțină un meniu drop-down realizat cu CSS;
- conținut generat (:after, :before);
- tranziție care implică schimbarea mai multor proprietăți la intervale diferite de timp;
- animație care să modifice mai multe proprietăți ale aceluiași element;
- conținutul siteului să aibă sens.

Cerințe JS (12 puncte)

- fișier separat JavaScript;
- modificarea stilului unui element sau al unui grup de elemente;
- manevrarea DOM-ului (selectare după id, tag, clasă, selectorCSS);
- creare și ștergere de elemente;
- folosirea și modificarea evenimentelor generate de mouse și tastatură;
- modificare de proprietăți;
- inputuri funcționale (de exemplu: input de tip text/range/number/radio/checkbox, select, textarea);
- setTimeout, setInterval;
- folosirea localStorage (să se păstreze în localStorage o colecție de elemente);

- folosirea a cel puțin unei metode din clasele: Math, Array, String, Date;
- schimbarea aleatoare a valorilor unor proprietăți (de exemplu: culoare, dimensiuni, poziții);
- folosirea proprietăților classList, target, currentTarget;
- folosirea metodelor getComputedStyle și stopPropagation cu un sens relevant;
- validarea datelor dintr-un formular folosind expresii regulate;
- elementele de JavaScript să fie integrate cu sens în cadrul proiectului

Cerințe Node.js

- cereri get/post: preluare date din formular
- cereri Ajax cu preluare date din fișiere json

Atenție!

Nu puteți folosi librării, framework-uri CSS sau JavaScript (cum ar fi jQuery, Bootstrap, Angular, React, etc) pentru realizarea frontend-ului.