

Tema laborator 1& 2: Proiectarea unei baze de date si analiza datelor

Acest document contine primele doua teme, care **se vor preda separat**. Enunturile sunt prezentate impreuna, ca sa puteti vedea faptul ca primele doua teme sunt legate una de alta, deci rezolvarea intr-un mod cat mai corect al primei teme va influenta si tema urmatoare.

Tema 1. Proiectare baza de date

Deadline: Saptamana 3 (tema e primita in saptamana 1)

Ganditi-va la o aplicatie simpla care are nevoie de o baza de date. Reprezentati datele aplicatiei intr-o structura relationala si implementati structura intr-o baza de date SQL Server. Baza de date trebuie sa contina **minim 7 tabele**, cel putin un tip de relatie **1:M** si cel putin un tip de relatie **M:N**.

Obs. Inainte sa incepeti rezolvarea laboratorului, sa va ganditi la subiectul pe care doriti sa il modelati (astfel incat sa poata fi extins la cel putin 7 tabele) si, eventual, va consultati cu coordonatorul laboratorului. In cazul in care exista un conflict cu tema propusa de un alt coleg, are castig de cauza cel care a ales primul tema respectiva. La link-ul urimator puteti consulta lista cu subiecte alese deja. Cand v-ati hotarat si ati verificat ca tema nu a fost deja aleasa, va treceti numele complet si subiectul pentru care doriti sa creati baza de date in tabel:

[Tabel Tema 1](#)

Tema 2. Populare baza de date si analiza date

Deadline: Saptamana 5 (tema e primita in saptamana 3)

Scrieti comenzi SQL pentru un tabel care sa:

- **insereze**,
- **modifice** si
- **stearga** date.

In comenzile de modificare sau stergere folositi cel putin o data in clauza WHERE: operatori logici, operatori relationali, **IS [NOT] NULL**.

Obs. Se pot utiliza *view*-uri in cel mult cinci interogari.

Atentie! Interogările trebuie să aibă sens în contextul bazei de date, să producă informații relevante pentru un potențial utilizator. Trebuie să puteți da un enunț pentru interogarea scrisă (de ex. afișăm toți studenții care au nota de trecere la o anumită materie). **Nu folosiți condiții pe Id-uri**, deoarece id-urile nu sunt relevante pentru utilizatorul bazei de date, de ex. nu voi spune că afișez toți studenții cu Id-ul mai mare decât 50 pentru că asta nu îmi spune nimic despre studenți, deci nu are sens pentru un utilizator.

Scrieti urmatoarele interogari SELECT:

- a. o interogare cu operatia de reuniune: cu UNION [ALL] sau OR;
- b. o interogare cu operatia de intersectie: cu INTERSECT sau IN;
- c. o interogare cu operatia de diferenta: cu EXCEPT sau NOT IN;
- d. o interogare cu INNER JOIN si o interogare cu unul dintre operatorii: LEFT JOIN, RIGHT JOIN, FULL JOIN; una dintre interogari va extrage date din trei tabele aflate in relatie *many-to-many*;
- e. o interogare care contine o subinterogare in clauza WHERE: cu IN sau EXISTS;
- f. o interogare care contine o subinterogare in clauza FROM;
- g. 2 interogari cu clauza GROUP BY; una dintre ele va contine si clauza HAVING; se vor folosi cel putin 2 operatori de agregare dintre: COUNT, SUM, AVG, MIN, MAX;

h. o interogare imbricata cu unul dintre operatorii ANY sau ALL, unde operatorul relational este din multimea {=, <, <=, >, >=, <>}.

In interogările de mai sus se vor folosi cel puțin o dată:

- expresii aritmetice in clauza SELECT;
- DISTINCT;
- ORDER BY;
- TOP