

## Baze de Date

### Informatică - Anul 1

## Cerințe proiect – semestrul 2, 2022-2023

Să se realizeze proiectarea, implementarea și exploatarea neprocedurală a unei baze de date relaționale, urmând pașii de mai jos:

1. Descrierea modelului real, a utilității acestuia și a regulilor de funcționare.
  2. Prezentarea constrângerilor (restricții, reguli) impuse asupra modelului.
  3. Descrierea entităților, incluzând precizarea cheii primare.
  4. Descrierea relațiilor, incluzând precizarea cardinalității acestora.
  5. Descrierea atributelor, incluzând tipul de date și eventualele constrângeri, valori implicite, valori posibile ale atributelor.
  6. Realizarea diagramei entitate-relație corespunzătoare descrierii de la punctele 3-5.
  7. Realizarea diagramei conceptuale corespunzătoare diagramei entitate-relație proiectate la punctul 6. Diagrama conceptuală obținută trebuie să conțină minimum 6 tabele (fără considerarea subentităților), dintre care cel puțin un tabel asociativ.
  8. Enumerarea schemelor relaționale corespunzătoare diagramei conceptuale proiectate la punctul 7.
  9. Realizarea normalizării până la forma normală 3 (FN1-FN3).
  10. Crearea unei secvențe ce va fi utilizată în inserarea înregistrărilor în tabele (punctul 11).
  11. Crearea tabelelor în SQL și inserarea de date coerente în fiecare dintre acestea (minimum 5 înregistrări în fiecare tabel neasociativ; minimum 10 înregistrări în tabelele asociative).
  12. Formulați în limbaj natural și implementați 5 cereri SQL complexe ce vor utiliza, în ansamblul lor, următoarele elemente:
    - subcereri sincronizate în care intervin cel puțin 3 tabele
    - subcereri nesincronizate în clauza FROM
    - grupări de date cu subcereri nesincronizate în care intervin cel puțin 3 tabele, funcții grup, filtrare la nivel de grupuri (în cadrul aceleiași cereri)
    - ordonări și utilizarea funcțiilor NVL și DECODE (în cadrul aceleiași cereri)
    - utilizarea a cel puțin 2 funcții pe șiruri de caractere, 2 funcții pe date calendaristice, a cel puțin unei expresii CASE
    - utilizarea a cel puțin 1 bloc de cerere (clauza WITH)
- Observație:** Într-o cerere se vor regăsi mai multe elemente dintre cele enumerate mai sus, astfel încât cele 5 cereri să le cuprindă pe toate.
13. Implementarea a 3 operații de actualizare și de ștergere a datelor utilizând subcereri.

14. Crearea unei vizualizări complexe. Dați un exemplu de operație LMD permisă pe vizualizarea respectivă și un exemplu de operație LMD nepermisă.
15. Formulați în limbaj natural și implementați în SQL: o cerere ce utilizează operația *outer-join* pe minimum 4 tabele, o cerere ce utilizează operația *division* și o cerere care implementează analiza top-*n*.

**Observație:** Cele 3 cereri sunt diferite de cererile de la exercițiul 12.

16. Optimizarea unei cereri, aplicând regulile de optimizare ce derivă din proprietățile operatorilor algebrei relaționale. Cererea va fi exprimată prin expresie algebrică, arbore algebric și limbaj (SQL), atât anterior cât și ulterior optimizării. **ALTERNATIVĂ: două instrucțiuni select echivalente semantic, de comparat din punct de vedere a execuției (explicat plan de execuție).**
17. a. Realizarea normalizării BCNF[, FN4, FN5].
- b. Aplicarea denormalizării, justificând necesitatea acesteia. **ALTERNATIVĂ: alegerea unor relații/join-uri din model și reprezentarea acestora într-o bază de date NoSql (MongoDb, Cassandra etc.)**
18. Tranzacții: ilustrarea *consistency levels* in Oracle cu tranzacții care operează asupra modelului ales. (model consistency\_level.sql)
19. Optimizarea a două cereri utilizând indexare.

## Observații

- Proiectul trebuie realizat **individual**.
- Proiectul trebuie realizat în sistemul **Oracle** (versiune minimă 11g) instalat pe calculatorul propriu.
- Un **proiect complex** presupune o diagramă conceptuală cu minimum 12 tabele (punctul 6), existența unei relații de tip superior (>2), tratarea complexă a fiecărei cerințe.
- Nu este necesară includerea tuturor atributelor în diagramele de la punctele 6 și 7; toate attributele vor fi enumerate obligatoriu în schemele relaționale (punctul 8).
- Denumirile tabelor și ale atributelor vor fi obligatoriu în limba română.
- Diagramele vor fi obligatoriu realizate folosind regulile și notațiile prezentate la curs.

- Diagrama conceptuală va fi proiectată în forma normală 3, astfel încât la punctul 9 să nu fie necesară o refacere a *design*-ului bazei de date, ci doar exemplificarea transformărilor de la fiecare formă normală, pe baza unor cazuri relative la model. Mai precis, se va da câte un exemplu de non-FN1, non-FN2, non-FN3 și transformarea fiecăruia în forma normală corespunzătoare. Este recomandabil ca rezultatul acestor normalizări să se regăsească în diagrama conceptuală (punctul 7).
- Pentru formele normale BCNF, FN4, FN5 se analizează modelul obținut la punctul 7 pentru a identifica relațiile care nu se află în aceste forme normale. Se va proceda într-unul dintre următoarele moduri:
  - Se aplică normalizarea în cazul identificat, fără a reproiecta modelul;
  - Se procedează asemănător observației anterioare (referitoare la FN1-FN3): se va da câte un exemplu de non-BCNF, non-FN4, non-FN5 și se vor aplica transformările pentru aducerea la BCNF, FN4, respectiv FN5. Exemplele vor fi relative la modelul proiectat anterior.

### Cerințe de redactare și condiții de eligibilitate:

Pentru a fi luat în considerare proiectul trebuie să conțină:

- Un fișier doc/docx/pdf care să integreze toate rezolvările cerințelor și care să îndeplinească următoarele condiții:
  - va fi structurat conform cerințelor proiectului, va avea pagină de titlu și cuprins generat;
  - va include *print-screen*-uri prin care să se demonstreze că tot codul inclus în proiect a fost rulat în *Oracle*;
  - va conține **tot codul SQL și sub formă de text** (nu doar ca imagine!);
  - pentru fiecare exercițiu va fi inclus enunțul în limbaj natural, codul SQL atât sub formă de text cât și ca *print screen* din SQL Developer cu codul sursă (complet sau parțial) și rezultatul execuției acestuia;
  - pentru problema 11 *print screen*-urile vor arăta că există date în tabelele create;
  - pentru fiecare exercițiu de la problema 12 se vor specifica elementele utilizate din lista prezentată în cadrul cerinței.
- Un fișier text care să conțină codul SQL pentru cerințele de la punctele 10-11 (comenzile de creare a secvenței, a tabelor și comenzile pentru inserarea datelor în aceste tabele);
- Un fișier text care să conțină codul SQL pentru cerințele de la punctele 12-13 (respectiv 12-15); se vor indica cerințele în limbaj natural și elementele SQL folosite sub formă de comentarii în cadrul acestui fișier, înaintea fiecărei implementări SQL.

Cele 3 fișiere de mai sus **vor fi denumite** conform următorului format: <grupa>\_<Nume>\_<Prenume>-<tip\_document>.<extensie>, unde *tip\_document* va avea valorile "proiect", "creare\_inserare", "exemple" (de exemplu: 131\_Popescu\_Ana-proiect.pdf, 131\_Popescu\_Ana-creare\_inserare.txt, 131\_Popescu\_Ana-exemple.txt). Fișierele astfel denumite vor fi încărcate până **la termenul limită stabilit (26 mai 2023, ora 23:59)**, *link*-ul pentru încărcare fiind anunțat pe Teams.