

Proiect

Proiectați și implementați folosind Oracle 11g/12c o bază de date relațională¹ (minim 6 entități independente și o tabelă asociativă).

Cerințe:

- a) Prezentarea concisă a bazei de date (utilizarea ei).
- b) Realizarea diagramei entitate-relație (ERD).
- c) Realizarea diagramei conceptuale pornind de la diagrama entitate-relație.
- d) Transformarea sistemului conceptual într-un design logic, subliniind relațiile dintre tabele, cheile primare și străine (externe).
- e) Transformarea design-ului logic într-un design fizic astfel încât sistemul rezultat la punctul d) să fie în FN3.
 - Dați un exemplu de atribut repetitiv (multivaloare) al unei entități din diagramă.
 - Dați un exemplu de tabel relațional din diagramă care este în FN1, dar nu în FN2. Să se aducă tabelul în FN2.
 - Dați un exemplu de tabel relațional din diagramă care este în FN2, dar nu în FN3. Să se aducă tabelul în FN3.
- f) Implementarea tabelelor în Oracle, folosind chei primare, constrângeri de referință și domenii. Adăugarea de informații coerente (minim 5 înregistrări pentru fiecare entitate independentă; minim 10 înregistrări pentru tabela asociativă).
- g) Scrierea a 15 interogări, cât mai complexe, care să illustreze **toate** aspectele învățate din lista (formulați în limbaj natural problemele ce urmează a fi rezolvate):
 - clauzele GROUP BY, HAVING, START WITH, CONNECT BY, ORDER BY;
 - funcții pentru lucrul cu șiruri de caractere și date calendaristice: LOWER, UPPER, SUBSTR, INSTR, TO_CHAR, TO_DATE, ADD_MONTHS, MONTHS_BETWEEN;
 - funcții diverse: DECODE, NVL, NULLIF, CASE);
 - INNER, LEFT, RIGHT, FULL JOIN;
 - operatori pe mulțimi;
 - funcții multiple-row/ agregat: AVG, SUM, MIN, MAX, COUNT;
 - subinterogări în clauzele: SELECT, FROM, WHERE, HAVING;
 - operatorul DIVISION.

¹ http://www.databasanswers.org/data_models/?fbclid=IwAR0B7IKsWhhEG2-qXAG7bmnmoUgrd5-KxYTks7u3ulMKOtDqXbyyLfhovM

h) Crearea un tabel de mesaje cu următoarea structură:

Field	Data type	Comments
Message_id	NUMBER	Cheie primară
Message	VARCHAR2(255)	
Message_type	VARCHAR2(1)	Valid values: E - Error, W - Warning, I - Information
Created_by	VARCHAR2(40)	NOT NULL
Created_at	DATE	NOT NULL

- i) Ilustrarea următoarelor noțiuni de PL / SQL (formulați în limbaj natural problemele ce urmează a fi rezolvate):
- subprogram stocat (inclusiv apelare) care să utilizeze 2 tipuri de colecție învățate;
 - subprogram stocat (inclusiv apelare) care să utilizeze un tip de cursor învățat;
 - subprogram stocat de tip funcție care să utilizeze într-o singură comandă SQL 3 dintre tabelele definite; tratarea tuturor excepțiilor care pot apărea; apelarea subprogramului astfel încât să fie evidențiate toate cazurile tratate;
 - Trigger de tip LMD la nivel de comandă (inclusiv declanșare);
 - Trigger de tip LMD la nivel de linie (inclusiv declanșare);
 - Trigger de tip LDD (inclusiv declanșare).
 - Pachet care să conțină toate obiectele definite în cadrul punctului i).
- j) În cazul obținerii unor excepții, dacă doriți să înregistrați unele avertismente sau alte informații, mesajele corespunzătoare vor fi inserate în tabelul MESAJE (id-ul mesajului se va insera automat folosind o secvență).