

PROIECT la DISCIPLINA
INTRODUCERE IN BAZE DE DATE
An academic 2020-2021
Punctaj – 30/100 puncte
Termen predare: saptamana 14/semI

I. SCOP si CERINTE GENERALE

Sa se realizeze o Bază de date si o aplicație care să gestioneze datele stocate în baza de date , corespunzător unui domeniu de activitate .

Pentru predarea proiectului va trebui sa pregatiti :

1. Aplicatia - arhiva cu cod si descriere Readme a acesteia si a modului de lansare în executie
2. Un Document (sursa si tiparit), care sa cuprinda conform unei structuri propuse (vezi Model) :
 1. tema proiectului – sumar executiv
 2. detalii de proiectare conceptuală a bazei de date (diagrama EER/UML)
 3. solutia de transformare în relational (daca este necesara, many to many, reflexivitate, alegere chei)
 4. descrierea bazei de date relationale – tabele (8 tabele /atribute/chei/indecsi)
 5. argumentarea nivelului de normalizare
 6. 20 interogari asupra datelor(fisier .txt), cod SQL si minim 5 interogări în algebra relationala
 7. Realizarea de obiecte de tip vedere(min 3), trigere (min 2), procedurile stocate (min 8), securizarea accesului
 8. implementarea a minim 2 formulare de acces la baza de date și cel puțin 2 rapoarte ce extrag date din mai mult de un tabel.
 9. descrierea aplicației din punct de vedere functional (module PHP/Java/Net) – descrierea minimala a fiecărei clase implementate si un model al ierarhiei de clase a proiectului
 10. manual de instalare/configurari /utilizare
 11. concluzii si dezvoltari ulterioare

Proiectul se va realiza în echipă de max . 2 studenți. Sunt admise solicitări de realizare a echipelor de proiect, cu membrii din grupe diferite, însă recomandarea este de a se realiza echipa în cadrul aceleiași grupe.

II. ETAPE in derularea proiectului

1.1.Implementarea schemei in SGBD.

Scop: identificarea structurii de date a aplicației, realizarea modelului abstract, conceptual în EER/UML. formularea cerintelor de utilizare pentru aplicatie

Cerinte asupra datelor si diagrama EER/UML: alegeti o aplicatie care sa cuprinda min 6-8 entitati si relatii (nontriviala).

1.2.Construirea , popularea si interogarea bazei de date

Scop: traduce diagrama EER/UML in scheme relationale, implementarea schemei intr-un SGBD: MSSQL/MySQL , popularea schemei BD.

Explicati modul in care ati realizat traducerea/reprezentarea in schema relationala. Pentru situatiile ce comporta posibil mai multe solutii argumentati alegerea.

Argumentati nivelul de normalizare a schemelor de relatii.

Scrieti un script SQL ce creeaza tabelele bazei de date, specificand tipurile atributelor, cheile primare, constrangeri not null si unique, valori default ,constrangeri de cheie straina si politicile necesare de reactie la modificari (ON DELETE NO ACTION este default), etc, eventual alte constrangeri suplimentare necesare si argumentate(tip check, sau trigger)

Incarcarea datelor. Scrieti un script suplimentar ce contine elemente INSERT pentru relatii, astfel incat constrangerile specificate sa nu fie violate.

Identificati in setul de interogari posibile si necesare, acelea care poseda urmatoarele caracteristici de implementare SQL : agregare cu group by si having, imbricare SELECT cu agregare, imbricare cu negare (NOT EXISTS si NOT IN) , joinuri pe aceeasi tabela/tabele diferite.

1.3. Vederi, proceduri, trigger. Se vor implementa pentru aplicatie numarul solicitat de componente programatice .

1.4.Integrarea elementelor de securitate si a minim doi utilizatori de sistem Implementarea interfetei aplicatiei.Se poate opta pentru implementare PHP/Java

III.STRUCTURĂ de PROIECT (se va utiliza modelul atasat)

UTCN

Facultatea de Automatica si Calculatoare

Departamentul de Calculatoare

Disciplina Baze de Date Anul xxxx

Denumire proiect (alegeti un acronim interesant)

Data

Echipa de proiect

Cuprins

1. Introducere

Introducere, argumente, scop si obiective specifice

2. Analiza cerintelor utilizatorilor (Specificatiile de proiect)

- Ipoteze specifice domeniului ales pentru proiect (cerinte, constrangeri)
- Organizare structurata(tabelar) a cerintelor utilizator
- Determinarea si caracterizarea de profiluri de utilizatori (admin, user , diversi alti “actori”)

3. Modelul de date si descrierea acestuia

- Entitati si attributele lor (descriere detaliata)
- Diagrama EER/UML pentru modelul de date complet

- Normalizarea datelor

4. Detalii de implementare

- Descrierea functională a modulelor (organizarea logică a acestora- de ex . structura claselor Java, module PHP)
- Elemente de utilizare/instalare (diferențiat pe tipuri de actori)
- Elemente de securizare a aplicației

5. Concluzii limitari si dezvoltari ulterioare

6. Bibliografie

Varianta 1

Teme libere pe care se vor imagina functionalitati diverse

Companie aeriana- gestiune zboruri
Olimpiade ntionale
Gestiune orar
Management restaurant/cafenea
BD nutritie
Biblioteca virtuala
Managementul stoduri librerie
Work&travel
Distributie Internet/telefonie/cablu tv
BD competitie Formula 1
BD inchirieri biciclete
Managementul unui imobil
BD pentru asigurari
Managementul activitati/personal/copii ai unei gradinite
Magazin online de cosmetice
Joc multiplayer
BD video on demand (seriale)
BD subtitrari filme
Gestiunea unei cantine studentesti
BD voluntariat
Managementul unui cabinet de medicina de familie
Online tickets
BD licitatie antichitati
Managementul unui centru donare sange
Managementul unui centru de echitatie
BD companie imobiliara
Managementul unei firme de organizare evenimente
BD campionat de baschet
Magazin online de biciclete si componente
Depozit de medicament
Magazin online de calculatoare si componente
Retea de socializare
Gestiunea unui complex de parcare
Management clienti si servicii provider de Internet
Gestiunea unui site de anuntiuri online
Managementul unei ligi de fotbal/competitie d etenis
Gestiunea fisierelor unei firme de traduceri
Managementul cazarilor in camine studentesti
Gestiunea unei florarii virtuale
Managementul unei firme de traduceri
Agentie de turism
Site de stiri
Managementul unei firme de securitate
Laborator de analize medicale
Managementul unui restaurant
Gestionarea resurselor umane dintr-o firma de outsourcing
Managementul unei firme de securitate
Agentie de turism
Managementul unei companii de inchirieri video
Magazin virtual de telefoane si accesorii
Managementul unei sali de fitness
Managementul unei companii de inchirieri auto

Modele de relaționale de date pot fi consultate la adresa : www.databaseanswers.org

Varianta 2 – Teme specificate

Este posibilă deasemenea alegerea unuia dintre proiectele propuse în specificațiile de mai jos.

Proiect BD1

Se considera un magazin virtual pentru vanzarea de produse electronice. Baza de date (considerata ca o colectie de tabele) trebuie sa contina informatiile enumerate mai jos.

- NumarCard
- Nume
- Prenume
- DataNasterii
- IdProdus
- Produs
- Garantie
- Stoc
- ValoareUnitara
- Cantitate
- Valoare unitară

Stiind ca ‘Numarul de Card’ are exact 8 caractere, ‘Numele’ nu depaseste 30 de caractere, ‘Prenumele’ nu depaseste 30 de caractere, descrierea unui produs nu depaseste 80 de caractere, în stoc nu vor fi nicodata mai mult de 200 de produse dintr-o anumită categorie, garantia nu poate sa fie mai mare de 5 ani, se cere:

- a) Să se realizeze proiectarea bazei de date aferente (structura de tabele, structura de coloane a fiecărei tabele, constrângeri):
 - i)Proiectare logică a bazei de date (forme normalizate) => structura de tabele;
 - ii)Dimensionare coloane, constrângeri;
- b) Să se scrie comenzile SQL pentru tabelele proiectate la punctul anterior.
- c) Să se scrie comenzile SQL pentru popularea bazei de date cu urmatoarele produse si cu urmatorii clienti:

IdProdus	Produs	Garantie	Stoc	ValoareUnitara
1	Fujitsu Siemens Amilo Pro	1	10	2000
2	Indesit WLI1000	3	5	900
3	Gorenje RC400	3	4	1500

NumarCont	Nume	Prenume	DataNasterii
11111111	Popescu	Ion	01-Jan-1985
22222222	Georgescu	Andreea	23- Aug -1983
33333333	Ionescu	Robert	08-Mar-1982

Fiecare client poate cumpara produse din mai multe categorii, la aceeasi data sau la date diferite. Pentru fiecare produs vandut, trebuie sa se salveze in baza de date, informatiile asociate vanzarii.

- d) Să se scrie un trigger care realizeaza decrementarea stocului atunci cand un produs a fost vandut.
- e) Să se scrie o procedură stocata care realizeaza vanzarea a unui anumit numar de produse catre acelasi client.

- f) Să se genereze un raport care sa afișeze toate cumparaturile facute de o persoana. Raportul va afișa următoarele coloane:

Nume
Prenume
Produs
Cantitate
ValoareTotala

- g) Să se genereze un raport care să afișeze toate produsele vandute pentru care nu a expirat inca garantia. Raportul va contine urmatoarele coloane:

Produs
DataExpirării

- h) Să se afișeze cel mai vandut produs;

- i) Să se afișeze data în care au avut loc cele mai multe vanzări;

- j) Să se afișeze clientul (nume,prenume) care a cumpărat cele mai multe produse (și valoarea totală a cumparaturilor).

Proiect BD2

Se considera o aplicatie online care este utilizata pentru realizarea bilantului contabil al unei firme. Aplicatia foloseste pentru stocarea datelor o baza de date (considerata ca o colectie de tabele) care trebuie sa contina informatiile enumerate mai jos.

- numar cont cinci cifre,
 - descriere 50 de caractere,
 - tip cont 2 caractere (ca –capital, pa-pasiv, ac-activ),
 - sold initial (suma disponibila in cont la inceputul anului curent),
 - sold (suma disponibila in cont la momentul curent),
 - numarul tranzactiei - nu poate fi mai mare decat 10000,
 - data tranzactiei (trebuie sa fie din anul curent),
 - suma tranzactionata - pentru un cont, intr-o singura tranzactie, nu poate fi mai mare decat 10000,
 - descrierea tranzactiei (maxim 10 caractere), Se cere:
1. Sa se scrie comenzile de creare a tabelelor, impunand toate constrangerile care sunt considerate utile din punct de vedere functional si al integritatii datelor.
 2. Sa se scrie comenzile de populare a tabelelor cu informatii (minim 5 articole in fiecare tabela).
 3. Sa se implementeze o procedura care sa implementeze o tranzactie, stiind ca aceasta presupune 2 conturi – unul debitor (din care pleaca banii) si respectiv unul creditor (in care intra banii), procedura primeste ca si parametrii de intrare numarul tranzactiei, contul debitor, contul creditor, suma tranzactionata precum si o descriere a tranzactiei.
 4. Sa se implementeze un trigger care atunci cand este introdusa o tranzactie automat sa calculeze soldul conturilor implicate in tranzactie
 5. Să se afișeze toate tranzacțiile în care este folosit un anumit cont.
 6. Să se afișeze toate tranzacțiile care au fost introduse în perioada (01.01.2010-01.06.2010)
 7. Să se calculeze și să se afișeze bilantul initial, suma totala creditoare, respectiv debitoare, pentru toate conturile
 8. Să se afișeze toate tranzacțiile care implică conturi de un anumit tip
 9. Să se steargă un cont daca nu exista tranzactii pentru el
 10. Să se afiseze contul care apare in cele mai multe tranzactii precum si numarul de tranzactii in care el apare.

Proiect BD3

Se considera o aplicatie pentru gestionarea mijloacelor fixe ale unei firme. Pentru stocarea datelor se foloseste o baza de date (considerata ca o colectie de tabele) care trebuie sa contina informatiile enumerate mai jos.

- Nr_inventar
- Descriere
- Data_punerii_in_serviciu
- Valoare_initiala
- Depreciere_speciala -
- Baza_de_depreciere
- Metoda_de_depreciere
- Nr_an
- An
- Depreciere_anterioara
- Depreciere_curenta

Stiind ca trebuie stocate urmatoarele informatii si ca acestea trebuie sa respecte constrangerile: nr inventar este mai mic decat 10000, descrierea are maxim 50 de caractere, valoarea initiala, deprecierea speciala, baza de depreciere, deprecierea anterioara, deprecierea curenta nu pot fi mai mari de 1000000, nr de ani in care se face deprecierea unui mijloc fix nu poate fi mai mare de 50 de ani, se cere:

- a) Să se realizeze proiectarea bazei de date aferente (structura de tabele, structura de coloane a fiecărei tabele, constrângeri):
 - i)Proiectare logică a bazei de date (forme normalizate) => structura de tabele;
 - ii)Dimensionare coloane, constrângeri;
 - b) Să se scrie comenzile SQL pentru tabelele proiectate la punctul anterior.
 - c) Sa se scrie comenzile SQL pentru prepopularea bazei de date cu urmatoarele doua metode de depreciere (Amortizare Lineara, Amortizare Accelerata).
 - d) Sa se implementeze un trigger care sa calculeze automat Baza de Depreciere, in momentul in care se introduce un nou mijloc fix (Baza de Depreciere = Valoarea Initiala – Deprecierea Speciala).
 - e) Sa se implementeze o procedura stocata care sa calculeze deprecierea pentru un anumit an (procedura va primi ca si parametru anul pentru care se calculeaza deprecierea si va introduce in baza de date o intrare de genul [AN, Depreciere Anterioara, Depreciere Curenta]). Deprecierea Anterioara reprezinta suma Deprecierilor Curente din anii anteriori anului pentru care se face calculul.
- Formulele de calcul a Deprecierii Curente pentru cele doua metode introduse la punctul (c) sunt:
- amortizare lineara: Depreciere Curenta = Baza de Depreciere / Nr. Ani.
 - Amortizare accelerata: Depreciere Curenta = (Baza de Depreciere – Deprecierea Anterioara) * 2 / Nr. Ani
- f) Sa se afiseze grupat in functie de metoda de depreciere toate mijloacele fixe, ordonat crescator dupa numarul de ani si data punerii in serviciu. Raportul va contine urmatoarele coloane:
 - Nr. Inventar
 - Descriere
 - Data Punerii in Serviciu
 - Metoda - Nr. Ani
 - Depreciere Anterioara (pentru data curenta)
 - Depreciere Curenta

- g) Sa se afiseze toate mijloacele fixe pentru care a expirat perioada de depreciere, ordonat dupa numarul de inventar
- h) Sa se scrie comenzile SQL pentru introducerea a trei mijloace fixe (perioade de depreciere diferite, iar doua mijloace sa aiba metode de depreciere diferite)
- i) Sa se afiseze toti anii de depreciere pentru un anumit mijloc fix (ordonat dupa an).
Raportul va contine:
- Nr. Inventar
 - Descriere
 - An
 - Depreciere Anterioara
 - Depreciere Curenta
- j) Sa se afiseze deprecierea totala pentru toate mijloacele fixe care folosesc metoda de depreciere Amortizare Lineara.

Proiect BD4

Se considera o aplicatie online pentru gestionarea unor jocuri precum sah, table etc. Pentru stocarea datelor se foloseste o baza de date (considerata ca o colectie de tabele) care trebuie sa contina informatiile enumerate mai jos.

- ID_jucator
- Nume
- Data_inscrierii
- Data_nasterii
- Id_joc
- Tip_joc
- Jucator1 (camp care contine Id-ul jucatorului 1)
- Jucator2 (camp care contine Id-ul jucatorului 2)
- Nr_partide
- Nr_partide_jucate
- Data_inceput_joc
- Data_sfarsit_joc
-
- Scor_juca
tor1 -
Scor_juca
tor2
- Invingator (Id-ul jucatorului care a castigat jocul)

Stiind ca aceste informatii trebuie sa respecte urmatoarele constrangeri: Nume maxim 30 de caractere, Data_nasterii si Data_inscrierii nu pot fi mai mari decat data curenta, Id_joc nu poate fi mai mare decat 1000000000, Tip_joc are maxim 20 de caractere, Nr_partide si Nr_partide_jucate nu pot fi mai mari decat 100, Scor_jucator1, Scor_jucator2 reprezinta numarul de partide castigate de fiecare jucator, Data_inceput_joc nu poate fi mai mare decat Data_sfarsit_joc, daca jocul nu s-a sfarsit Data_sfarsit_joc este NULL, Data_inceput_joc nu poate fi nula, daca jocul nu s-a terminat invingatorul este NULL, un joc nu poate fi jucat de mai mult de doi jucatori, se cere:

- a) Să se realizeze proiectarea bazei de date aferente (structura de tabele, structura de coloane a fiecărei tabele, constrângeri):
 - i)Proiectare logică a bazei de date (forme normalizate) => structura de tabele;
 - ii)Dimensionare coloane, constrângeri;
- b) Să se scrie comenzile SQL pentru tabelele proiectate la punctul anterior.
- c) Sa se scrie comenzile SQL pentru popularea bazei de date cu urmatoarele doua jocuri: sah si table.
- d) Sa se scrie o procedura stocata care creaza un nou joc, procedura primeste ca si parametru tipul jocului, numele celor doi jucatori si numarul de partide, numarul de partide trebuie sa fie impar.
- e) Sa se implementeze un trigger care sa seteze Invingatorul si Data de Sfarsit a unui joc daca Nr Partide a devenit egal cu NrPartide Jucate
- f) Sa se afiseze toate jocurile jucate si incheiate intre 01.ian.2010 si 01.apr.2010, sortate crescator dupa tipul jocului si dupa data de inceput. Raportul va contine:
 - Id Joc
 - Tip Joc
 - Nr. Partide
 - Data Inceput_joc

- Data Sfarsit_joc
- Durata Joc (in ore)
- Invingator

g) Sa se afiseze cel mai bun jucator de sah.

h) Sa se afiseze jucatorul care a participat la cele mai multe jocuri.

i) Sa se afiseze pentru fiecare jucator jocurile la care acel jucator a participat. Raportul va contine:

- Nume
- Id Joc
- Tip Joc
- Data_inceput_joc
- Data_sfarsit_joc
- Castigator (,da' - daca a castigat jocul, ,nu' - in caz contrar)

j) Sa se afiseze cel mai bun jucator pe categorii de varsta (<10, 10-18, 18-40, 40-50, >50);

Proiect BD12

Se considera o aplicatie pentru evidenta clientilor care achizitioneaza produse de la o firma distribuitoare. Pentru orice Client trebuie sa se salveze intr-o baza de date urmatoarele informatii:

- nume
- prenume
- CNP
- adresa
- telefon
- disponibil in cont (suma de bani achitata de client in avans) - informatii despre achizitiile efectuate:
 - Produs
 - Data achizitiei
 - Pret
 - Suma incasata

Stiind ca 'Numele' nu depaseste 15 caractere, 'Prenumele' nu depaseste 20 caractere, 'CNP' are exact 13 caractere, telefonul contine doar cifre si este exact de 9 caractere, disponibil in cont este pozitiv si cu 2 zecimale, si ca un client poate avea oricate achizitii, se cere:

1. Să se realizeze proiectarea bazei de date aferente (structura de tabele, structura de coloane a fiecărei tabele, constrângeri).
2. Sa se scrie comenzile SQL pentru tabelele proiectate la punctul anterior.
3. Să se scrie comenzile SQL pentru popularea bazei de date cu urmatoarele produse:

nume	prenume	CNP	adresa	telefon	Disponibil	produs	data	pret	Suma incasata
Ion	Adi	123456	Timis	123456789	0	Indesit	10/01/2007	66	66
Achim	Gheorghe	123457	Resita	234567890	0	Siemens	01/02/2003	100	100
Dima	Alex	123456	Arad	326774434	0	Bosch	07/12/2005	120	20
Duma	Mihai	123456	Deva	576325767	0	LG	11/12/2005	55	37
Ion	Adi	123456	Timis	123456789	0	Sony	14/05/2006	7.99	1.99

4. Să se scrie o procedura care sa permita vanzarea unui produs (La vanzarea unui produs, se va verifica disponibilul clientului, si acesta va fi adaugat sumei incasate. In cazul in care suma incasata depaseste valoarea ramasa de achitat, diferenta va fi trecuta in disponibilul clientului) - procedura va fi apelata cu parametrii: CNP, nume produs, data, pret, suma incasata.
5. Sa se genereze un raport care sa cuprinda numele, prenumele, cnp, si produsele achizitionate si achitate integral de catre clienti.
6. Sa se genereze un raport detaliat care sa cuprinda numele, prenumele, CNP, si toate produsele achizitionate, pretul, si restul de plata, ordonat dupa nume, prenume, data achitiei, crescator, respectiv rest de plata descrescator.
7. Sa se scrie un trigger care la adaugarea unei produs, sa calculeze automat disponibilul si suma incasata, in conditiile enuntate la punctul 4.
8. Sa se scrie o functie care sa primeasca ca si parametri CNP, si produs sa returneze restul de plata.

9. Sa se afiseze doar clientii care au minim 4 produse achitate integral, sau au achitionat produse a caror valoare totala depaseste 1000, in maxim 2 ani consecutivi precizand: nume, prenume, CNP si respectiv suma totala incasata., si care nu au produse achitate partial (Clienti majori).
10. Sa se afiseze clientul care are cele mai multe produse achizitionate dar neplatite integral, precizand numele, prenumele, CNP, numar de produse si rata lui de achitare (cat la suta a platit din pretul bunurilor achizitionate).

Proiect BD6

Se considera o aplicatie pentru evidenta parcului auto. Baza de date va contine urmatoarele informatii:

- Numar inmatriculare
- Marca
- An fabricatie
- carburant
- data alimentarii
- kilometraj – cand s-a facut alimentarea
- soferul care a facut alimentarea

Stiind ca 'Numarul' are exact 7 caractere, 'marca' nu depaseste 10 caractere, anul de fabricatie este cuprins intre 2000 si 2007, carburant este numarul de litri de combustibil de la alimentare (nu poate fi mai mult de 60), kilometraj este un numar intreg, care contine numarul de km efectuati de la ultima realimentare, si soferul este de maxim 10 caractere, si ca un o masina poate fi realimentata de oricate ori, dar numai umpland la maxim rezervorul, se cere:

1. Să se realizeze proiectarea bazei de date aferente (structura de tabele, structura de coloane a fiecărei tabele, constrângeri).
2. Sa se scrie comenzile SQL pentru tabelele proiectate la punctul anterior.
3. Să se scrie comenzile SQL pentru popularea bazei de date cu urmatoarele produse:

numar	marca	An fabricatie	carburant	Data alimentarii	kilometraj	sofer
TMXXYY1	Logan	2007	40	10/01/2007	100	Duma A.
TMXXYY2	Renault	2006	34	01/02/2003	50	Ion M.
TMXXYY3	Aro	2000	35	07/12/2005	150	Pop A.
TMXXYY4	Cielo	2001	55	11/12/2005	230	Popa M.
TMXXYY5	Matiz	2005	34	14/05/2006	89	Marin O.
TMXXYY1	Logan	2007	40	10/02/2007	180	Duma A.

OBS: se considera ca initial toate masinile au rezervorul plin.

4. Să se scrie o procedura care sa permita alimentarea unei masini - procedura va fi apelata cu parametrii: numar, carburant, data, kilometraj, sofer.
5. Sa se genereze un raport care sa cuprinda numarul, anul fabricatiei, si consumul mediu al tuturor masinilor din parc.
6. Sa se genereze un raport detaliat care sa cuprinda numarul, anul fabricatiei, data alimentarii, soferul si consumul, ordonat dupa sofer, consum, numar, in sens crescator si data alimentarii in sens descrescator.
7. Sa se scrie un trigger care sa nu permita adaugarea informatiilor despre alimentari, daca consumul mediu rezultat este de 3 ori mai mic decat consumul mediu total pe masina respectiva si sa semnaleze eroare.
8. Sa se scrie o functie care sa primeasca ca si parametri sofer, si sa returneze consumul mediu.
9. Sa se afiseze primele trei masini cu cele mai multe alimentari, in 2 ani consecutivi precizand: numar, an fabricatie, marca, carburant, data alimentarii.

10. Sa se afiseze soferul care are cel mai mare consum general (calculat pentru tot kilometrajul facut), precizand pentru fiecare masina condusa de respectivul sofer urmatoarele: sofer, numar, consum, si ponderea kilometrajului cu respectiva masina in numarul total de km parcursi de respectivul sofer.

Se considera o aplicatie pentru monitorizarea evenimentelor unei firme de paza. Baza de date Oracle va contine urmatoarele informatii:

- Nume client
- CNP
- Adresa
- Eveniment (descriere)
- De plata
- data producerii evenimentului
- agentul care a constatat evenimentul
- timpul de reactie la eveniment (in cat timp a ajuns agentul la fata locului) - timp mediu de reactie pe agent.

Stiind ca Nume client are maxim 14 caractere, CNP are exact 13 caractere, adresa are maxim 10 caractere, eveniment are maxim 20 caractere, agentul este de maxim 10 caractere, de plata este un numar cu 2 zecimale, pozitiv, timpul de reactie este pozitiv si exprimat in secunde, se cere:

1. Să se realizeze proiectarea bazei de date aferente (structura de tabele, structura de coloane a fiecărei tabele, constrângeri).
2. Sa se scrie comenzile SQL pentru tabelele proiectate la punctul anterior.
3. Să se scrie comenzile SQL pentru popularea bazei de date cu urmatoarele produse, cu 4 clienti, 2 avand cate 3 evenimente, unul un eveniment si altul nici un eveniment. Evenimentele sunt toate constatate de agenti diferiti.
4. Să se scrie o procedura care sa permita adaugarea unui eveniment - procedura va fi apelata cu parametrii: CNP, eveniment, data, agent si timp de reactie. Procedura va actualiza si suma de plata a clientului, stiind ca un eveniment pe an este gratuit, si ca restul costa 10 daca timpul a fost sub 300 secunde si 5 altfel.
5. Sa se genereze un raport care sa cuprinda numele, CNP si numarul total de evenimente, precum si timpul mediu de reactie al tuturor agentilor pentru acele evenimente.
6. Sa se genereze un raport detaliat care sa cuprinda numele, CNP, eveniment, data, agent, timp ordonat dupa nume, agent in sens crescator si timp in sens descrescator.
7. Sa se scrie un trigger care dupa adaugarea unui eveniment, sa actualizeze suma de plata.
8. Sa se scrie o functie care sa primeasca ca si parametri CNP, si sa returneze timpul mediu de reactie.
9. Sa se afiseze primii doi clienti cu cele mai multe evenimente, in 2 ani consecutivi precizand: nume, CNP, adresa, eveniment, agent.
10. Sa se afiseze agentul cu cele mai multe evenimente, precizand agent, eveniment, adresa, si timp de reactie precizand ponderea timpului de reactie pe fiecare caz in timpul total de reactie al agentului.

Proiect la BD7

Se considera o societate de creditare, pentru persoane fizice si juridice. Datele sunt structurate in urmatoarele tabele:

- **clienti_fizice** avand urmatoarele campuri: ID, nume, prenume, telefon, CNP, adresa. Tabela cuprinde atat clientii persoane fizice cat si reprezentantii firmelor care sunt stocati in tabela **clienti_juridice**.
- **clienti_juridice** avand urmatoarele campuri: ID, nume, telefon, CUI, adresa, ID_reprez. O firma poate avea un singur reprezentant.
- **credite** avand urmatoarele campuri: IDcredit, IDclient, tip_client, suma, perioada, comision, dobanda, achitat. Un client poate avea mai multe credite pe numele sau.
- **plati** avand urmatoarele campuri: IDplata, IDcredit, data, suma. Pentru un credit se pot face oricate plati.
- **garantii** avand urmatoarele campuri: IDgarantie, IDcredit, descriere, suma, tip. Un credit poate avea mai multe garantii. O garantie nu poate fi utilizata la mai multe credite.

Cerinte:

1. Sa se scrie comenzile de creare a tabelelor, impunand toate constrangerile care sunt considerate utile din punct de vedere functional si al integritatii datelor.
2. Sa se scrie comenzile de populare a tabelelor cu informatii (minim 5 articole in fiecare tabela).
3. Sa se afiseze totalul de bani aflat pe piata, sub forma:

Clienti	Plasament
Persoane fizice xxxxxxxxx	Persoane juridice xxxxxxxxx
Total	xxxxxxxxx

4. Sa se afiseze resursele financiare disponibile ale societatii (banii care pot fi oferiti spre creditare clientilor) stiind ca initial societatea avea un capital de 1.000.000.
5. Stiind ca reprezentantii unei firme sunt stocati in tabela **clienti_fizice**, indiferent daca au credite pe umele lor sau nu, legatura fiind asigurata de coloana ID_reprez, sa se scrie interogarea care afiseaza o lista cu toti clientii persoane fizice care au credite atat pe numele lor, cat si pe firmele a caror reprezentanti sunt.
6. Sa se scrie interogarea care afiseaza toate firmele care au aceiasi reprezentanti, si au credite, grupate pe reprezentanti. Afisarea va include: nume reprezentant, prenume reprezentant, firma, CUI, suma creditata, suma achitata.
7. Stiind ca fiecare garantie este stocata in tabela garantii, si ca un credit poate avea mai multe garantii, sa se afiseze lista creditelor care au garantii, grupat pe numarul de garantii.
8. Stiind ca tipul garantiilor specificate in tabela **garantii** poate fi: mobil, imobil si cec, plecand de la aceasta tabela sa se creeze o tabela **garant_comb** avand urmatoarele campuri: IDcredit, tip_garantie, suma. Unde tip garantie este *mobil*, daca toate garantiile creditului sunt mobile, *imobil*, daca toate garantiile creditului sunt imobile, cec, daca toate garantiile creditului sunt cecuri si respectiv *combinat* daca creditul are garantii de diferite tipuri.
9. Considerand tabela de la punctul 8 creata, sa se scrie un trigger care la orice modificare a tabelii garantii, actualizeaza automat si tabela garant_comb.

10. Sa se scrie un trigger care la orice modificare in tabela plati sa actualizeze coloana **achitat** in tabela credite. Se considera o aplicatie pentru evidenta calculatoarelor si a licentelor software dintr-o firma. Pentru orice angajat trebuie sa se salveze intr-o baza de date Oracle urmatoarele informatii:

- nume
- prenume
- nr. Legitimatie
- descriere computer
- nr. Inventar (pentru computer) - informatii despre licentele software:
 - tipul de licenta (OEM, retail, OPEN, Inchiriere, free, other)
 - Produs
 - Producator

- Valoare de achizitie
- Document de achizitie

Stiind ca ‘Numele’ nu depaseste 15 caractere, ‘Prenumele’ nu depaseste 20 caractere, ‘Numarul de Legitimatie’ are exact 6 caractere, descrierea unui produs nu depaseste 40 caractere, un angajat poate avea maxim 5 calculatoare si ca pe un computer pot fi instalate oricate soft-uri, se cere:

1. Să se realizeze proiectarea bazei de date aferente (structura de tabele, structura de coloane a fiecărei tabele, constrângeri).
2. Sa se scrie comenzile SQL pentru tabelele proiectate la punctul anterior.
3. Să se scrie comenzile SQL pentru popularea bazei de date cu urmatoarele produse:

nume	prenume	legitimatie	Descriere computer	Nr. inventar	Tip licenta	produs	producator	valoare	document
popa	ion	123456	Pentium 4	1000	OEM	Wndows XP	Microsoft	500	34/20.01.2006
adam	gheorghe	123457	Athlon	1001	Retail	Office XP	Microsoft	1200	234/11.01.2007
popa	ion	123456	Pentium 4	1000	Free	openOffice	XXX	0	NA
popa	ion	123456	Celeron	1002	OPEN	Windows 2000	Microsoft	400	23/02.02.2003
Pop	george	123458	Celeron	1003	Free	7zip	XXX	0	NA

4. Să se scrie o procedura care sa aloce un calculator unui angajat. (se va apela procedura cu 2 parametri: nr. legitimatie si nr.inventar).
5. Sa se genereze un raport care sa cuprinda numarul de licente, valoarea totala a acestora in functie de producator.
6. Sa se genereze un raport detaliat care sa cuprinda numarul de licente, valoarea totala a acestora in functie de producator, tip licenta si anul achizitiei, ordonat dupa an in sens descrescator, apoi dupa producator si tip licenta. Anul este reprezentat de ultimele 4 caractere din coloana document de achizitie.
7. Sa se scrie un trigger care la stergerea unui calculator din baza de date sa elimine toate licentele de tip OEM care au fost alocate acelui calculator.
8. Sa se scrie o functie care sa primeasca ca si parametru numele produsului si sa returneze valoarea totala a acestuia.
9. Sa se afiseze calculatoarele si utilizatorii acestora care nu au alocate produsele Windows sau Office, precizand: nr. inventar, nume, prenume, nr. legit si respectiv ce produs lipseste.
10. Sa se afiseze utilizatorul care utilizeaza licente avand cea mai mare valoare totala de achizitie (indiferent de numarul de calculatoare din dotare).

Proiect BD8

Se considera o aplicatie pentru evidenta cheltuielilor unei firme, care efectueaza plati in valuta (registru de casa). Baza de date Oracle contine urmatoarele informatii:

- curs valutar: data, rata de schimb
- parteneri: nume, CUI, Adresa
- tranzactii: data efectuarii, partenerul, suma in valuta, suma in lei - bilant zilnic: data, sold initial, total intrari, total iesiri, sold final.

Stiind ca rata de schimb este cuprinsa intre 1.00 si 4.00, numele nu depaseste 10 caractere, CUI are exact 6 caractere, suma este negativa daca se plateste partenerului si pozitiva daca se incaseaza de la partener, si ca soldul initial a fost 0, se cere:

1. Să se realizeze proiectarea bazei de date aferente (structura de tabele, structura de coloane a fiecărei tabele, constrângeri).
2. Sa se scrie comenzile SQL pentru tabelele proiectate la punctul anterior.
3. Să se scrie comenzile SQL pentru popularea bazei de date cu 5 tranzactii efectuate considerand 3 parteneri in 2 zile consecutive.
4. Să se implementeze un mecanism care sa nu permita modificarea cursului valutar pentru zilele cand sunt efectuate tranzactii, sa se argumenteze alegerea facuta.
5. Sa se scrie o procedura care sa genereze un raport care sa cuprinda numele,CUI, intrari, iesiri pentru partenerul specificat sa si argument prin CUI.
6. Sa se genereze un raport detaliat care sa cuprinda numele,CUI, si toate tranzactiile efectuate (data, tip tranzactie (intrare sau iesire), suma in valuta), ordonat dupa nume si anul efectuarii tranzactiilor.
7. Sa se scrie un trigger care la adaugarea unei tranzactii sa calculeze automat suma in lei, iar daca cursul nu este specificat sa semnaleze eroare si sa nu permita adaugarea tranzactiei.
8. Sa se scrie o functie care sa primeasca ca si parametru CUI si sa returneze rulajul efectuat de acel partener (suma dintre intrari si iesiri luate fara semn).
9. Sa se afiseze partenerii care doar au incasat si care nu au facut plati intr-un an intreg, precizand: nume, CUI, suma incasata , an.
10. Sa se afiseze partenerul care are cele mai multe tranzactii, precizand numele, CUI, numar de tranzactii, rata lui de rentabilitate (cat la suta este profit din suma totala a tranzactiilor).