

GESTIUNEA UNEI COMPANII DE COSMETICE

BERLINSCHI ALEXANDRA MARINA
GRUPA 142

CERINTA 1

Descrierea modelului real, a utilității acestuia și a regulilor de funcționare.

Compania de cosmetice produce și distribuie produse cosmetice de calitate superioară în întreaga lume. Compania are mai multe linii de produse, inclusiv îngrijirea pielii, machiaj, parfumuri și produse de îngrijire personală.

Compania are o echipă de cercetare și dezvoltare care dezvoltă și testează noi produse, iar aceste produse sunt testate pe un grup de voluntari înainte de a fi lansate pe piață. Compania deține propriile laboratoare pentru producția produselor cosmetice și are parteneri externi pentru a livra produsele la clienți.

Compania are un departament de marketing care se concentrează pe strategii de branding și promovare pentru a își promova produsele. Acest departament colaborează cu departamentul de vânzări pentru a găsi noi piețe și pentru a crește vânzările existente.

Baza de date relatională va fi utilizată pentru a gestiona toate informațiile legate de produse, cercetare și dezvoltare, producție, distribuție, vânzări și marketing. Acesta va permite companiei să își gestioneze eficient toate aspectele operaționale ale afacerii.

Regulile de funcționare:

- Toate produsele trebuie să fie testate înainte de a fi lansate pe piață. Fiecarui produs i se va atribui data în care a fost testat și un calificativ.
- Fiecare produs va avea un cod propriu, o categorie de care va aparține, un preț de fabricare și un preț de vânzare. De asemenea, acesta va avea review din partea clienților.
- Echipa de cercetare va avea un reprezentant și un director. De asemenea, aceasta va avea un număr de cercetări lunare și o rată a reușitei produselor din punct de vedere al respectării normelor de sănătate.
- Fiecare grup de voluntari va avea un cod unic, o medie a eșecului în vederea reacțiilor alergice și o medie a feedback-ului pentru produsul testat.
- Echipa de marketing va avea un lider. Pentru fiecare strategie de marketing va fi atribuit un cod care reprezintă șansele de reușită. De asemenea, fiecare echipă va

avea o rata a reusitei in ceea ce priveste promovarea produsului si un cost mediu al promovarii.

- Fiecare client va avea un cont propriu cu datele personale : nume, prenume, adresa, numar de telefon, adresa de mail. Fiecarui client i se va atribui un cod unic si un cod care ii contorizeaza comenzile astfel incat recomandarile pot fi redistribuite in functie de preferinte.
- Pentru comenzile online exista un cod unic atribuit fiecare comenzi. De asemenea, se va contoriza data la care a fost efectuata comanda si starea comenzii.
- O comanda poate fi efectuata de catre mai multi parteneri externi din apropiere pentru obtinerea unui discount , aceasta urmand sa fie expediat separate

CERINTA 2

Prezentarea constrângerilor (restricții, reguli) impuse asupra modelului

Restricții de functionare:

- Compania de cosmetice are cel puțin un produs cosmetic distribuit
- Fiecare produs este testat
- Un voluntar trebuie să testeze cel puțin un produs
- Compania are cel puțin un client
- Clientul trebuie să aibă date valide și nenule : număr de telefon, email, nume, prenume
- O comandă trebuie să conțină minim un produs
- Un produs trebuie să fie testat cel puțin o dată înainte de a ajunge pe piață și , de asemenea, are nevoie de un calificativ bun
- O comandă are un cod unic atribuit pentru contorizarea în baza de date
- O recomandare va fi transmisă către minim un client
- Compania poate avea Parteneri Externi astfel încât poate ajunge cu produsele în mai multe locuri
- Comenzile însemnate vor fi făcute către partenerii externi astfel încât acestea conțin un număr foarte mare de produse
- Un laborator trebuie să aibă cel puțin o echipă cu cel puțin un om
- În timpul unui eveniment compania trebuie să aibă reduceri pentru minim un produs
- O promovare trebuie să fie făcută de minim o echipă de marketing
- Departamentul de Cercetare și Dezvoltare trebuie să aibă minim un laborator
- Pretul de fabricare și pretul de vânzare este calculate per comandă

CERINTA 3

Descrierea entităților, incluzând precizarea cheii primare.

Pentru baza de date referitoare la Gestiunea unei Companii de Cosmetice am următoarele Entități : Produse, CercetareDezvoltare, Voluntari, Marketing, Clienti, Comenzi, Laboratoare, ParteneriExterni, Eveniment, Promovare, Recomandare, Comenzilnsemnate

Produse = Stocheaza informatii despre fiecare produs fabricat si distribuit de companie. Cheia primara este ID_Produs

CercetareDezvoltare = Reprezinta echipa de cercetare și dezvoltare din cadrul companiei. Este responsabila cu dezvoltarea și testarea noilor produse. Cheia primara este ID_Reprezentant

Voluntari = Reprezinta echipa de voluntari responsabila cu testarea. Ajuta la evaluarea si monitorizarea eficacitatii si sigurantei produselor inainte de lansarea lor pe piata. Cheia primara este ID_Voluntar

Marketing = Reprezinta echipa de marketing a companiei. Are rolul de a dezvolta si implementa strategii de branding si promovare pentru a creste vanzarile produselor. Cheia primara este ID_Lider

Clienti = Stocheaza informatii despre clientii companiei. Cheia primara este ID_Client

Comenzi = Reprezinta comenzile plasate de catre client. Este utilizata pentru gestionarea si urmarirea comenzilor efectuate de client. Cheia primara este ID_Comanda

Laboratoare = Stocheaza informatii despre laboratoarele detinute de companie. Este utilizata pentru gestionarea si monitorizarea activitatilor de productie cosmetice in propriile laboratoare ale companiei. Cheia primara este ID_Laborator

ParteneriExterni = Reprezinta partenerii externi ai companiei , care se ocupa de livrarea si vanzarea produselor catre client. Este responsabila de gestionarea si monitorizarea relatiilor cu partenerii externi ai companiei. Cheia primara este ID_Partener

Eveniment = Se refera la evenimentele speciale organizate de companie, in cadrul carora vor fi oferite reduceri. Permite gestionarea si monitorizarea evenimentelor si a reducerilor asociate . Cheia primara este ID_Eveniment

Promovare = Reprezinta informatii despre activitatile de promovare ale produselor. Este utilizata pentru gestionarea si urmarirea activitatilor de promovare desfasurate de catre echipa de marketing. Cheia primara este ID_Promovare

Recomandare = Stocheaza recomandarile facute de client cu privier la produsele companiei. Ajuta la gestionarea si monitorizarea feedback-ului pozitiv si a recomandarilor primite. Cheia primara este ID_Recomandare

Comenzilnsemnate = Reprezinta comenzile care necesita o atentie speciala si sunt semnalate pentru partenerii externi. Este utilizata pentru gestionarea comenzilor care necesita o cantitate mare de produse si o atentie suplimentara. Cheia primara este ID_ComandaInsemnata

Ingrediente = Stocheaza informatiile cu privire la ingredientele produselor. Cheia primara este ID_Ingredient

Stocuri = Stocheaza informatiile cu privire la stocurile companiei. Cheia primara este ID_Stoc

CERINTA 4

Descrierea relațiilor, incluzând precizarea cardinalității acestora.

COMENZI_sunt efectuate_CLIENTI : Aceasta este o relatie de tip one-to-many intre entitatile CLIENTI si COMENZI. Aceasta indica faptul ca un client poate face mai multe vanzari, dar o vanzare este facuta de un singur client. Cardinalitatea minima este de 1:0 (un client poate sa nu faca nicio vanzare) si cardinalitatea maxima este 1:n

PRODUSE_apartine_COMENZI : Aceasta este o relatie de tip one-to-many intre entitatile PRODUSE si COMENZI. Aceasta indica faptul ca un produs poate apartine mai multor comenzi. Cardinalitatea minima este 1:0 (un produs poate sa nu apartina unei comenzi) si cardinalitatea maxima este 1:n

VOLUNTARI_apartin_CERCETAREDEZVOLTARE: Aceasta este o relatie de tip many-to-one intre entitatile VOLUNTARI si CERCETAREDEZVOLTARE. Aceasta indica faptul ca mai multi voluntari pot apartine unui departament de cercetare si dezvoltare. Cardinalitatea minima este m:1 si cardinalitatea maxima este 1:n

LABORATOARE_apartin_CERCETAREDEZVOLTARE : Aceasta este o relatie de tip many-to-one intre entitatile LABORATOARE si CERCETAREDEZVOLTARE. Aceasta indica faptul ca mai multi voluntari pot apartine unui departament de cercetare si dezvoltare. Cardinalitatea minima este m:1 si cardinalitatea maxima este 1:n

EVENIMENT_este organizat _MARKETING : Aceasta este o relatie de tip many-to-one intre entitatile EVENIMENT si MARKETING. Aceasta indica faptul ca mai multe evenimente pot fi organizate de departamentul de marketing. Cardinalitatea minima este m:0 si cardinalitatea maxima este 1:n

PROMOVARE_este facuta_MARKETING : Aceasta este o relatie de tip many-to-one intre entitatile PROMOVARE si MARKETING. Aceasta indica faptul ca mai multe promovari pot fi organizate de departamentul de marketing. Cardinalitatea minima este m:0 si cardinalitatea maxima este 1:n

PARTENERIEXTERNI_ia legatura_MARKETING: Aceasta este o relatie de tip many-to-one intre entitatile PARTENERIEXTERNI si MARKETING. Aceasta indica faptul ca mai multi parteneri externi sunt organizati (pentru comenzi) de departamentul de marketing . Cardinalitatea minima este m:0 si cardinalitatea maxima este 1:n

PRODUSE_este transmis_CLIENTI : Aceasta este o relatie de tip many-to-many intre entitatile PRODUSE si CLIENTI, reprezentata prin tabelul asociativ RECOMANDARE. Aceasta indica faptul ca produsele pot ajunge la mai multi client, iar clientii pot “ajunge” la mai multe produse prin recomandari. Cardinalitatea minima este 0:n si cardinalitatea maxima este m:n

PARTENERIEXTERNI_efectueaza_COMENZI : Aceasta este o relatie de tip many-to-many intre entitatile PARTENERIEXTERNI si COMENZI, reprezentata prin tabelul asociativ COMENZIINSEMNATE. Aceasta indica faptul ca partenerii externi pot efectua mai multe comenzi, iar comenzile pot fi efectuate de catre mai multi parteneri externi prin comenzi insemnate (cumpararea pachetelor care contin mai multe comenzi in acelasi timp) . Cardinalitatea minima este 0:n si cardinalitatea maxima este m:n

CERINTA 5

Descrierea atributelor, incluzând tipul de date și eventualele constrângeri, valori implicite, valori posibile ale atributelor.

- Entitatea Comenzi are ca attribute :
 - ID_Comanda – variabila de tip intreg (PRIMARY KEY) care reprezinta codul unic al comenzii
 - ID_Client – variabila de tip intreg care reprezinta codul unic al clientului
 - DataComanda – variabila de tip data calendaristica care reprezinta data la care a fost efectuata comanda
 - StareComanda – variabila de tip caracter, de lungime maxima 50 care reprezinta starea actuala a comenzii
 - ID_Client int – variabila de tip int (FOREIGN KEY) care reprezinta codul unic al clientului

- Entitatea Clienti are ca attribute :
 - ID_Client – variabila de tip intreg (PRIMARY KEY) care reprezinta codul unic al clientului
 - Nume – variabila de tip caracter, de lungime maxima 50, care reprezinta numele clientului
 - Prenume – variabila de tip caracter, de lungime maxima 50, care reprezinta prenumele clientului
 - Adresa – variabila de tip caracter, de lungime maxima 100, care reprezinta adresa clientului
 - NrTelefon – variabila de tip caracter, de lungime maxima 20, care reprezinta numarul de telefon al clientului
 - Email – variabila de tip caracter, de lungime maxima 100, care reprezinta email-ul clientului

- Entitatea Produse are ca attribute :
 - ID_Produs – variabila de tip int (PRIMARY KEY) care reprezinta codul unic al produsului
 - NumeProdus – variabila de tip caracter, de lungime maxima 100, care reprezinta numele produsului
 - Categorie – variabila de tip caracter, de lungime maxima 50, care reprezinta categoria produsului
 - PretFabricare – variabila de tip numar zecimal care reprezinta pretul fabricarii produsului
 - PretVanzare – variabila de tip numar zecimal care reprezinta pretul vanzarii produsului
 - Review – variabila de tip caracter, de lungime maxima 255, care reprezinta review-ul produsului
 - ID_Comanda – variabila de tip int (FOREIGN KEY) care reprezinta codul unic al comenzii

- Entitatea Recomandare are ca attribute :
 - ID_Recomandare – variabila de tip int (PRIMARY KEY) care reprezinta codul unic al recomandarii
 - ID_Client – variabila de tip int (FOREIGN KEY) care reprezinta codul unic al clientului
 - ID_Produs – variabila de tip int (FOREIGN KEY) care reprezinta codul unic al produsului
 - DataRecomandare – variabila de tip data calendaristica care reprezinta data la care a fost trimisa recomandarea

- Entitatea Marketing are ca attribute :
 - ID_Echipe – variabila de tip int (PRIMARY KEY) care reprezinta codul unic al echipei
 - ID_Strategie - variabila de tip int care reprezita numarul strategiei aplicate
 - RataReusita – variabila de tip numar zecimal care reprezinta rata reusitei
 - CostMediuPromovare – variabila de tip numar zecimal care reprezinta costul mediu pe care compania l-a suportat pentru promovare

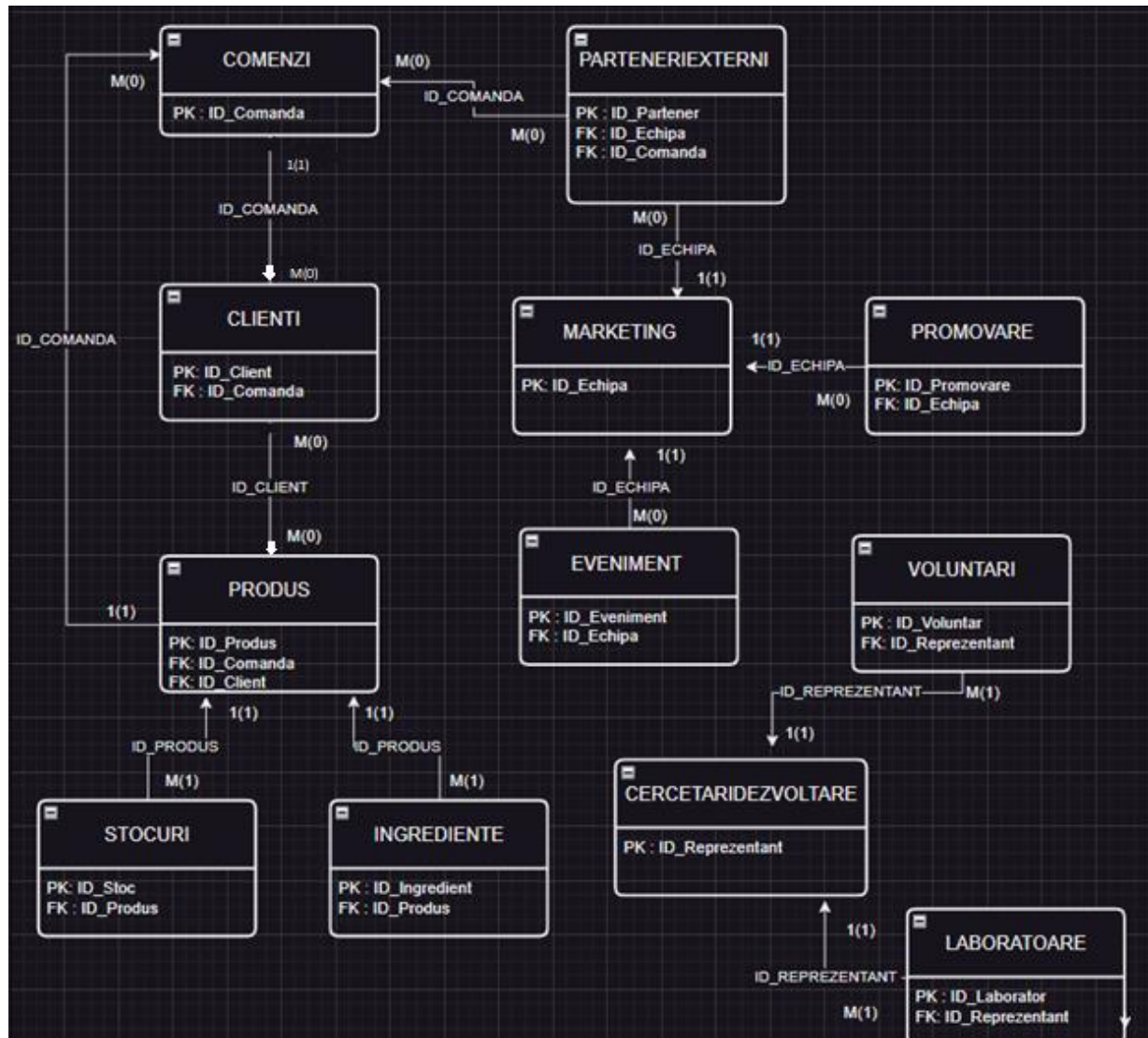
- Entitatea ParteneriExterni are ca attribute:
 - ID_Partener – variabila de tip int (PRIMARY KEY) care reprezinta codul unic al partenerului
 - NumePartener – variabila de tip caracter, de lungimea maxima 100, care reprezinta numele partenerului
 - Adresa – variabila de tip caracter, de lungime maxima 50, care reprezinta adresa sediului partenerului
 - Contact – variabila de tip caracter, de lungime maxima 50, care reprezinta numarul de contact al partenerului
 - ID_Echipa – variabila de tip int (FOREIGN KEY) care reprezinta codul unic al echipei de marketing
 - ID_ComandaInsemnata – variabila de tip int (FOREIGN KEY) care reprezinta codul unic al comenzii insemnate
- Entitatea ComenziInsemnate are ca attribute :
 - ID_ComandaInsemnate – variabila de tip int (PRIMARY KEY) care reprezinta codul unic al comenzii insemnate
 - ID_Comanda – variabila de tip int (FOREIGN KEY) care reprezinta codul unic al comenzii
 - ID_MotivInsemnare – variabila de tip caracter , de lungimea maxima 255, care reprezinta motivul insemnarii comenzii
 - ID_Partener – variabila de tip int care reprezinta codul unic al partenerului
- Entitatea Eveniment are ca attribute :
 - ID_Eveniment – variabila de tip int (PRIMARY KEY) care reprezinta codul unic al evenimentului
 - NumeEveniment – variabila de tip caracter, de lungime maxima 100, care reprezinta numele evenimentului
 - Descriere – variabila de tip caracter, de lungime maxima 255, care reprezinta descrierea evenimentului
 - DataInceput – variabila de tip data calendaristica care reprezinta data de inceput a evenimentului
 - DataSfarsit – variabila de tip data calendaristica care reprezinta data de sfarsit a evenimentului

- ProcentReducere – variabila de tip numar zecimal care reprezinta procentul de reducere pe care echipa de marketing il efectueaza produselor
 - ID_Echipa – variabila de tip int (FOREIGN KEY) care reprezinta codul unic al echipei
- Entitatea Promovare are ca attribute :
 - ID_Promovare – variabila de tip int (PRIMARY KEY) care reprezinta codul unic al promovarii
 - ID_Produs – variabila de tip int care reprezinta codul unic al produsului
 - DataPromovare – variabila de tip data calendaristica care reprezinta data campaniei de promovare
 - ID_Echipa – variabila de tip int (FOREIGN KEY) care reprezinta codul unic al echipei
- Entitatea CercetareDezvoltare are ca attribute :
 - ID_Reprezentant – variabila de tip int (PRIMARY KEY) care reprezinta codul reprezentantului echipei de cercetare
 - ID_Director – variabila de tip int care reprezinta codul directorului echipei de cercetare
 - NumarCercetariLunar – variabila de tip int care reprezinta numarul de cercetari al echipei
 - RataReusita – variabila de tip numar zecimal care reprezinta rata reusitei proiectului de cercetare
- Entitatea Voluntari are ca attribute:
 - ID_Voluntar – variabila de tip int (PRIMARY KEY) care reprezinta codul voluntarului
 - MedieEsecAlargii – variabila de tip numar zecimal care reprezinta media esecului voluntarului din punct de vedere al alergiilor
 - MedieFeedback – variabila de tip numar zecimal care reprezinta media feedback-ului oferit
 - ID_Reprezentant – variabila de tip int (FOREIGN KEY) care reprezinta codul reprezentantului

- Entitatea Laboratoare are ca attribute :
 - ID_Laboratoare – variabila de tip int (PRIMARY KEY) care reprezinta codul unic al laboratorului
 - NumeLaborator – variabila de tip caracter, de lungime maxima 100, care reprezinta numele laboratorului
 - AdresaContact – variabila de tip caracter, de lungime maxima 100, care reprezinta adresa laboratorului
 - Contact – variabila de tip caracter, de lungime maxima 50, care reprezinta contactul laboratorului
 - ID_Reprezentant – variabila de tip int (FOREIGN KEY) care reprezinta codul unic al reprezentantului departamentului de cercetare si dezvoltare
- Entitatea Ingrediente are ca attribute:
 - ID_Ingrediente – variabila de tip int (PRIMARY KEY) care reprezinta codul unic al ingredientului
 - NumeIngredient – variabila de tip caracter , de lungime maxima 100, care reprezinta numele ingredientului
 - Descriere – variabila de tip caracter, de lungime maxima 255, care reprezinta descriere ingredientului
 - ID_Produs – variabila de tip int (FOREIGN KEY) care reprezinta codul unic al produsului
- Entitatea Stocuri are ca attribute :
 - ID_Stoc – variabila de tip int (PRIMARY KEY) care reprezinta codul unic al stocului
 - Cantitate – variabila de tip int care reprezinta cantitatea
 - DataActualizare – variabila de tip data calendaristica care reprezinta data actualizarii
 - ID_Produs – variabila de tip int (FOREIGN KEY) care reprezinta codul unic al produsului

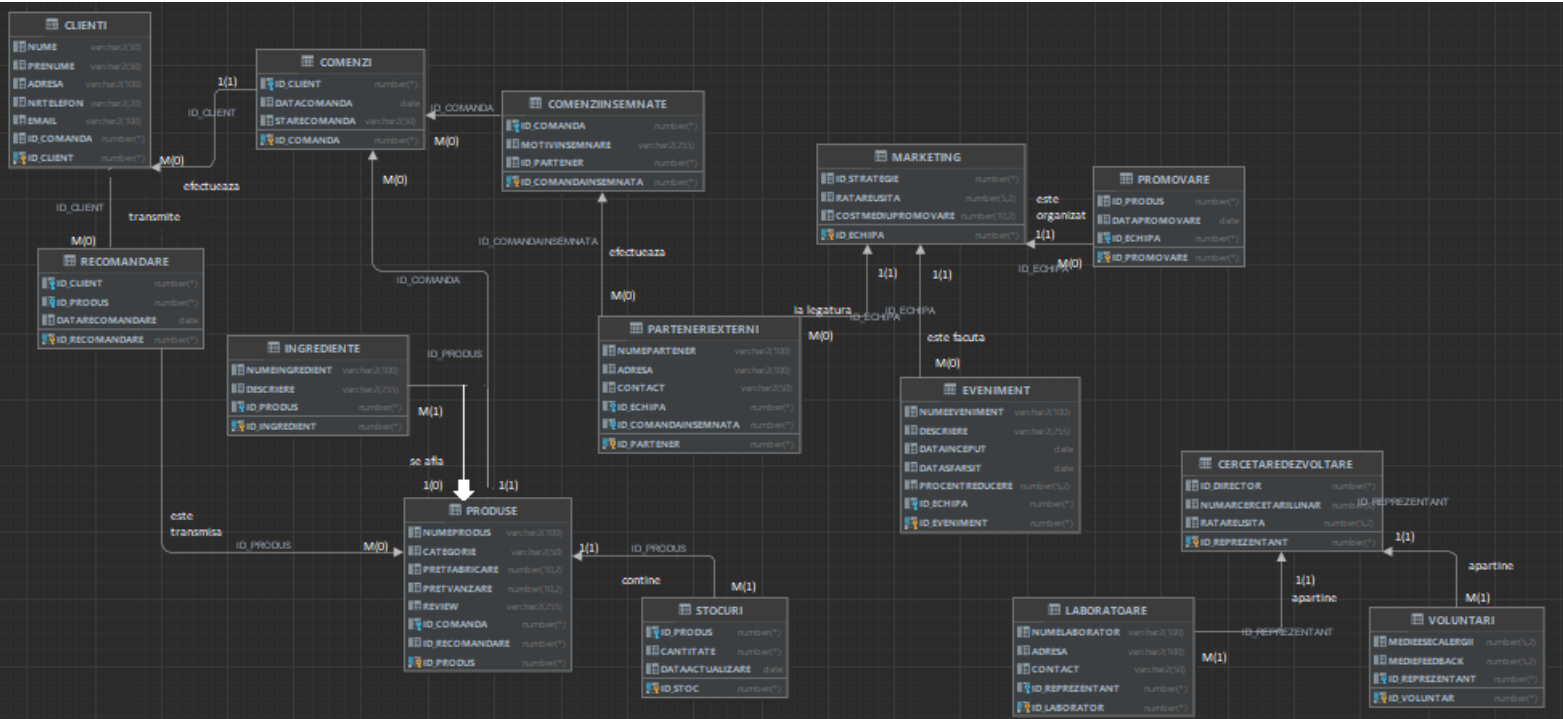
CERINTA 6

Realizarea diagramei entitate-relație corespunzătoare descrierii de la punctele 3-5



CERINTA 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.



CERINTA 8

Enumerarea schemelor relaționale corespunzătoare diagramei conceptuale proiectate la punctul 7.

1. Comenzi (ID_Comanda#, ID_Client#, DataComanda, StareComanda)
2. Clienti (ID_Client#, Nume, Prenume, Adresa, NrTelefon, Email, ID_Comanda)
3. Produse (ID_Produs#, NumeProdus, Categorie, PretFabricare, PretVanzare, Review, ID_Comanda#, ID_Recomandare)
4. Recomandare (ID_Recomandare#, ID_Client#, ID_Produs#, DataRecomandare)
5. Marketing (ID_Echipa#, ID_Strategie, RataReusita, CostMediuPromovare)
6. ParteneriExterni (ID_Partener#, NumePartener, Adresa, Contact, ID_Echipa#, ID_ComandaInsemnata#)
7. ComenzilInsemnate (ID_ComandaInsemnata#, ID_Comanda#, MotivInsemnare, ID_Partener)
8. Eveniment (ID_Eveniment#, NumeEveniment, Descriere, DataInceput, DataSfarsit, ProcentReducere, ID_Echipa#)
9. Promovare (ID_Promovare#, ID_Produs, DataPromovare, ID_Echipa#)
10. CercetareDezvoltare (ID_Reprezentant # , ID_Director, NumarCercetariLunar, RataReusita)
11. Voluntari (ID_Voluntar#, MedieEsecAlergii , MedieFeedback, ID_Reprezentant#)
12. Laboratoare (ID_Laborator#, NumeLaborator, Adresa, Contact, ID_Reprezentant#)
13. Ingrediente (ID_Ingrediente#, NumeIngredient, Descriere, ID_Produs#)
14. Stocuri (ID_Stoc#, ID_Produs#, Cantitate, DataActualizare)

CERINTA 9

Realizarea normalizării până la forma normală 3 (FN1-FN3)

- **FORMA NORMALA 1**

Daca fiecarui atribut ii corespunde o valoare indivizibila, atunci stim ca relatia este in prima forma normala. Deci puteti vedea ca toate relatiile din diagrama urmeaza aceasta conventie.

ID_ComandaInsemnata	ID_Comanda
100	1, 2 , 3
101	5, 4, 1
102	2, 3, 4

O varianta de normalizare este:

ID_ComandaInsemnata#	ID_Comanda#
101	1
102	2
103	3
104	5
105	4
106	1
107	2
108	3
109	4

- FORMA NORMALA 2

Daca o relatie este in forma 1-normala, atunci stim ca relatia este in forma 2-normala. Toate attributele care nu sunt chei primare sau nu contribuie la cheile primare depind de intreaga cheie primara. Retineti ca diagrama se incadreaza in forma normala 2

ID_Produs	NumeProdus	Categorie	PretFabricare	PretVanzare	Review	ID_Comanda	ID_Recomandare
1	Rimel	Machiaj	10.00	20.00	Bun	1	1
2	Spf	Ingrijirea pielii	30.00	70.00	Excelent	3	6

Se pot face urmatoarele dependente :

{ID_Produs# } -> { NumeProdus, Categorie }

{ ID_Comanda# } -> { PretFabricare, PretVanzare }

{ ID_Recomandare# } -> {Review }

Obtinem asadar :

ID_Produs	NumeProdus	Categorie
1	Rimel	Machiaj
2	SPF	Ingrijirea pielii

ID_Comanda	PretFabricare	PretVanzare
1	10.00	20.00
2	30.00	70.00

ID_Recomandare	Review
1	Bun
6	Excelent

- FORMA NORMALA 3

Daca o relatie este in forma normala 2 și toate attributele fara cheie (sau care nu participă la cheie) sunt direct dependente de cheia primara, atunci stim ca relatia este in forma normală 3.

Fie schema relationala Eveniment (ID_Eveniment#, NumeEveniment, Descriere, adresa, oras, ProcentReducere) . Pentru relatia Eveniment sunt dependentele : {ID_Eveniment } -> {NumeEveniment, Descriere, ProcentReducere, ID_Echipa# } si {ID_Locatie } -> { adresa, oras }

Astfel relatia Eveniment este in forma normala 2, dar nu este in forma normala 3.

Pentru a obtine o relatie in forma normala 3 se aplica regula CaseyDelobel si relatia Eveniment se proiecteaza in doua relatii .

Eveniment { ID_Eveniment#, NumeEveniment, Descriere, ProcentReducere, ID_Echipa#}

Perioada { ID_Perioada#, DataInceput, DataSfarsit }

CERINTA 10

Crearea unei secvențe ce va fi utilizată în inserarea înregistrărilor în tabele (punctul 11)

Pentru tabelul Comenzi am creat secventa Comenzi_nr :

```
CREATE SEQUENCE Client_nr
  START WITH 1
  INCREMENT BY 1
  MAXVALUE 5
  NOCYCLE;
```

```
CREATE SEQUENCE Comenzi_nr
  START WITH 1
  INCREMENT BY 1
  MAXVALUE 5
  NOCYCLE;
```

Pentru tabelul Clienti am creat secventa Client_nr:

```
CREATE SEQUENCE Comenzi_nr
  START WITH 1
  INCREMENT BY 1
  MAXVALUE 5
  NOCYCLE;
```

```
CREATE SEQUENCE Client_nr
  START WITH 1
  INCREMENT BY 1
  MAXVALUE 5
  NOCYCLE;
```

CERINTA 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).

- Tabelul Comenzi:

```
• CREATE TABLE Comenzi (  
  ID_Comanda INT PRIMARY KEY,  
  ID_Client INT,  
  DataComanda DATE,  
  StareComanda VARCHAR(50),  
  FOREIGN KEY (ID_Client) REFERENCES Clienti (ID_Client)  
);
```

```
CREATE TABLE Comenzi (  
  ID_Comanda INT PRIMARY KEY,  
  ID_Client INT,  
  DataComanda DATE,  
  StareComanda VARCHAR(50),  
  FOREIGN KEY (ID_Client) REFERENCES Clienti (ID_Client)  
);
```

```
CREATE SEQUENCE Comenzi_nr  
  START WITH 1  
  INCREMENT BY 1  
  MAXVALUE 5  
  NOCYCLE;  
  
INSERT INTO Comenzi (ID_Comanda, ID_Client, DataComanda, StareComanda)  
VALUES (Comenzi_nr.NEXTVAL, 1, TO_DATE('2021-05-25','YYYY-MM-DD'), 'In curs  
de procesare');  
  
INSERT INTO Comenzi (ID_Comanda, ID_Client, DataComanda, StareComanda)  
VALUES (Comenzi_nr.NEXTVAL, 2, TO_DATE('2021-05-25','YYYY-MM-DD'), 'In curs  
de procesare');  
  
INSERT INTO Comenzi (ID_Comanda, ID_Client, DataComanda, StareComanda)  
VALUES (Comenzi_nr.NEXTVAL, 3, TO_DATE('2021-05-26','YYYY-MM-DD'), 'In curs  
de procesare');  
  
INSERT INTO Comenzi (ID_Comanda, ID_Client, DataComanda, StareComanda)  
VALUES (Comenzi_nr.NEXTVAL, 4, TO_DATE('2021-05-27','YYYY-MM-DD'), 'In curs
```

```
de procesare');  
  
INSERT INTO Comenzi (ID_Comanda, ID_Client, DataComanda, StareComanda)  
VALUES (Comenzi_nr.NEXTVAL, 5, TO_DATE('2021-05-27','YYYY-MM-DD'), 'In curs  
de procesare');
```

```
CREATE SEQUENCE Comenzi_nr  
  START WITH 1  
  INCREMENT BY 1  
  MAXVALUE 5  
  NOCYCLE;  
  
INSERT INTO Comenzi (ID_Comanda, ID_Client, DataComanda, StareComanda)  
VALUES (Comenzi_nr.NEXTVAL, 1, TO_DATE('2021-05-25','YYYY-MM-DD'), 'In curs de procesare');  
  
INSERT INTO Comenzi (ID_Comanda, ID_Client, DataComanda, StareComanda)  
VALUES (Comenzi_nr.NEXTVAL, 2, TO_DATE('2021-05-25','YYYY-MM-DD'), 'In curs de procesare');  
  
INSERT INTO Comenzi (ID_Comanda, ID_Client, DataComanda, StareComanda)  
VALUES (Comenzi_nr.NEXTVAL, 3, TO_DATE('2021-05-26','YYYY-MM-DD'), 'In curs de procesare');  
  
INSERT INTO Comenzi (ID_Comanda, ID_Client, DataComanda, StareComanda)  
VALUES (Comenzi_nr.NEXTVAL, 4, TO_DATE('2021-05-27','YYYY-MM-DD'), 'In curs de procesare');  
  
INSERT INTO Comenzi (ID_Comanda, ID_Client, DataComanda, StareComanda)  
VALUES (Comenzi_nr.NEXTVAL, 5, TO_DATE('2021-05-27','YYYY-MM-DD'), 'In curs de procesare');
```

	ID_COMANDA	ID_CLIENT	DATAComanda	STARECOMANDA
1	1	1	2021-05-25	In curs de procesare
2	2	2	2021-05-25	In curs de procesare
3	3	3	2021-05-26	In curs de procesare
4	4	4	2021-05-27	In curs de procesare
5	5	5	2021-05-27	In curs de procesare

- Tabelul Clienti :

```
CREATE TABLE Clienti (
  ID_Client INT PRIMARY KEY,
  Nume VARCHAR(50),
  Prenume VARCHAR(50),
  Adresa VARCHAR(100),
  NrTelefon VARCHAR(20),
  Email VARCHAR(100),
  ID_Comanda INT,
);
```

```
CREATE TABLE Clienti (
  ID_Client INT PRIMARY KEY,
  Nume VARCHAR(50),
  Prenume VARCHAR(50),
  Adresa VARCHAR(100),
  NrTelefon VARCHAR(20),
  Email VARCHAR(100),
  ID_Comanda INT
);
```

```
CREATE SEQUENCE Client_nr
  START WITH 1
  INCREMENT BY 1
  MAXVALUE 5
  NOCYCLE;

INSERT INTO Clienti ( ID_Client, Nume, Prenume, Adresa, NrTelefon, Email,
ID_Comanda )
VALUES (1, 'Popescu', 'Ana', 'Str. Florilor, Nr. 10', '0721122334',
'ana.popescu@gmail.com', 1);

INSERT INTO Clienti ( ID_Client, Nume, Prenume, Adresa, NrTelefon, Email,
ID_Comanda )
VALUES (2, 'Ionescu', 'Maria', 'Str. Florilor, Nr. 11', '0721122335',
'ionescu.maria@gmail.com', 2);

INSERT INTO Clienti ( ID_Client, Nume, Prenume, Adresa, NrTelefon, Email,
ID_Comanda )
VALUES (3, 'Popa', 'Andrei', 'Str. Florilor, Nr. 12', '0721122336',
'popa.andrei@gmail.com', 3);

INSERT INTO Clienti ( ID_Client, Nume, Prenume, Adresa, NrTelefon, Email,
ID_Comanda )
VALUES (4, 'Pop', 'Andreea', 'Str. Florilor, Nr. 13', '0721122337',
'pop.andreea@gmail.com', 4);
```

```
INSERT INTO Clienti ( ID_Client, Nume, Prenume, Adresa, NrTelefon, Email,
ID_Comanda )
VALUES (5, 'Popescu', 'Andrei', 'Str. Florilor, Nr. 14', '0721122338',
'popescu.andrei@yahoo.com', 5);
```

```
INSERT INTO Clienti ( ID_Client, Nume, Prenume, Adresa, NrTelefon, Email, ID_Comanda )
VALUES (1, 'Popescu', 'Ana', 'Str. Florilor, Nr. 10', '0721122334', 'ana.popescu@gmail.com', 1);

INSERT INTO Clienti ( ID_Client, Nume, Prenume, Adresa, NrTelefon, Email, ID_Comanda )
VALUES (2, 'Ionescu', 'Maria', 'Str. Florilor, Nr. 11', '0721122335', 'ionescu.maria@gmail.com', 2);

INSERT INTO Clienti ( ID_Client, Nume, Prenume, Adresa, NrTelefon, Email, ID_Comanda )
VALUES (3, 'Popa', 'Andrei', 'Str. Florilor, Nr. 12', '0721122336', 'popa.andrei@gmail.com', 3);

INSERT INTO Clienti ( ID_Client, Nume, Prenume, Adresa, NrTelefon, Email, ID_Comanda )
VALUES (4, 'Pop', 'Andreea', 'Str. Florilor, Nr. 13', '0721122337', 'pop.andreea@gmail.com', 4);

INSERT INTO Clienti ( ID_Client, Nume, Prenume, Adresa, NrTelefon, Email, ID_Comanda )
VALUES (5, 'Popescu', 'Andrei', 'Str. Florilor, Nr. 14', '0721122338', 'popescu.andrei@yahoo.com', 5);
```

	ID_CLIENT	NUME	PRENUME	ADRESA	NRTELEFON	EMAIL	ID_COMANDA
1	1	Popescu	Ana	Str. Florilor, Nr. 10	0721122334	ana.popescu@gmail.com	1
2	2	Ionescu	Maria	Str. Florilor, Nr. 11	0721122335	ionescu.maria@gmail.com	2
3	3	Popa	Andrei	Str. Florilor, Nr. 12	0721122336	popa.andrei@gmail.com	3
4	4	Pop	Andreea	Str. Florilor, Nr. 13	0721122337	pop.andreea@gmail.com	4
5	5	Popescu	Andrei	Str. Florilor, Nr. 14	0721122338	popescu.andrei@yahoo.com	5

- Tabelul Produse :

```
• CREATE TABLE Produse (  
    ID_Produs INT PRIMARY KEY,  
    NumeProdus VARCHAR(100),  
    Categorie VARCHAR(50),  
    PretFabricare DECIMAL(10, 2),  
    PretVanzare DECIMAL(10, 2),  
    Review VARCHAR(255),  
    ID_Comanda INT,  
    ID_Recomandare INT,  
    FOREIGN KEY (ID_Comanda) REFERENCES Comenzi (ID_Comanda)  
);
```

```
CREATE TABLE Produse (  
    ID_Produs INT PRIMARY KEY,  
    NumeProdus VARCHAR(100),  
    Categorie VARCHAR(50),  
    PretFabricare DECIMAL(10, 2),  
    PretVanzare DECIMAL(10, 2),  
    Review VARCHAR(255),  
    ID_Comanda INT,  
    ID_Recomandare INT,  
    FOREIGN KEY (ID_Comanda) REFERENCES Comenzi (ID_Comanda)  
);
```

```
INSERT INTO Produse (ID_Produs, NumeProdus, Categorie, PretFabricare,  
PretVanzare, Review, ID_Comanda, ID_Recomandare)  
VALUES (1, 'Rimel', 'Machiaj', 10.00, 20.00, 'Bun', 1, 1);  
  
INSERT INTO Produse (ID_Produs, NumeProdus, Categorie, PretFabricare,  
PretVanzare, Review, ID_Comanda, ID_Recomandare)  
VALUES (2, 'Ruj', 'Machiaj', 15.00, 30.00, 'Foarte bun', 2, 2);  
  
INSERT INTO Produse (ID_Produs, NumeProdus, Categorie, PretFabricare,  
PretVanzare, Review, ID_Comanda, ID_Recomandare)  
VALUES (3, 'Crema de fata', 'Ingrijirea pielii', 10.00, 20.00, 'Bun', 3, 3);  
  
INSERT INTO Produse (ID_Produs, NumeProdus, Categorie, PretFabricare,  
PretVanzare, Review, ID_Comanda, ID_Recomandare)  
VALUES (4, 'Fond de ten', 'Machiaj', 12.00, 25.00, 'Foarte bun', 4, 4);  
  
INSERT INTO Produse (ID_Produs, NumeProdus, Categorie, PretFabricare,  
PretVanzare, Review, ID_Comanda, ID_Recomandare)  
VALUES (5, 'Gel de curatare ', 'Ingrijirea tenului', 15.00, 30.00,  
'Excelent', 5, 5);
```

```

INSERT INTO Produse (ID_Produs, NumeProdus, Categorie, PretFabricare,
PretVanzare, Review, ID_Comanda, ID_Recomandare)
VALUES (6, 'SPF', 'Ingrijirea pielii', 30.00, 70.00, 'Excelent', 3, 6);

INSERT INTO Produse (ID_Produs, NumeProdus, Categorie, PretFabricare,
PretVanzare, Review, ID_Comanda, ID_Recomandare)
VALUES (7, 'Pasta de dinti', 'Ingrijirea dentara', 5.00, 12.00, 'Excelent',
1, 7);

```

```

INSERT INTO Produse (ID_Produs, NumeProdus, Categorie, PretFabricare, PretVanzare, Review, ID_Comanda, ID_Recomandare)
VALUES (1, 'Rimel', 'Machiaj', 10.00, 20.00, 'Bun', 1, 1);

INSERT INTO Produse (ID_Produs, NumeProdus, Categorie, PretFabricare, PretVanzare, Review, ID_Comanda, ID_Recomandare)
VALUES (2, 'Ruj', 'Machiaj', 15.00, 30.00, 'Foarte bun', 2, 2);

INSERT INTO Produse (ID_Produs, NumeProdus, Categorie, PretFabricare, PretVanzare, Review, ID_Comanda, ID_Recomandare)
VALUES (3, 'Crema de fata', 'Ingrijirea pielii', 10.00, 20.00, 'Bun', 3, 3);

INSERT INTO Produse (ID_Produs, NumeProdus, Categorie, PretFabricare, PretVanzare, Review, ID_Comanda, ID_Recomandare)
VALUES (4, 'Fond de ten', 'Machiaj', 12.00, 25.00, 'Foarte bun', 4, 4);

INSERT INTO Produse (ID_Produs, NumeProdus, Categorie, PretFabricare, PretVanzare, Review, ID_Comanda, ID_Recomandare)
VALUES (5, 'Gel de curatare ', 'Ingrijirea tenului', 15.00, 30.00, 'Excelent', 5, 5);

INSERT INTO Produse (ID_Produs, NumeProdus, Categorie, PretFabricare, PretVanzare, Review, ID_Comanda, ID_Recomandare)
VALUES (6, 'SPF', 'Ingrijirea pielii', 30.00, 70.00, 'Excelent', 3, 6);

INSERT INTO Produse (ID_Produs, NumeProdus, Categorie, PretFabricare, PretVanzare, Review, ID_Comanda, ID_Recomandare)
VALUES (7, 'Pasta de dinti', 'Ingrijirea dentara', 5.00, 12.00, 'Excelent', 1, 7);

```

	ID_PRODUS	NUMEPRODUS	CATEGORIE	PRETFABRICARE	PRETVANZARE	REVIEW	ID_COMANDA	ID_RECOMANDARE
1	1	Rimel	Machiaj	10.00	20.00	Bun	1	1
2	2	Ruj	Machiaj	15.00	30.00	Foarte bun	2	2
3	3	Crema de fata	Ingrijirea pielii	10.00	20.00	Bun	3	3
4	4	Fond de ten	Machiaj	12.00	25.00	Foarte bun	4	4
5	5	Gel de curatare	Ingrijirea tenului	15.00	30.00	Excelent	5	5
6	6	SPF	Ingrijirea pielii	30.00	70.00	Excelent	3	6
7	7	Pasta de dinti	Ingrijirea dentara	5.00	12.00	Excelent	1	7

- Tabelul Recomandare :

```
• CREATE TABLE Recomandare (  
    ID_Recomandare INT PRIMARY KEY,  
    ID_Client INT,  
    ID_Produs INT,  
    DataRecomandare DATE,  
    FOREIGN KEY (ID_Client) REFERENCES Clienti (ID_Client),  
    FOREIGN KEY (ID_Produs) REFERENCES Produse(ID_Produs)  
);
```

```
CREATE TABLE Recomandare (  
    ID_Recomandare INT PRIMARY KEY,  
    ID_Client INT,  
    ID_Produs INT,  
    DataRecomandare DATE,  
    FOREIGN KEY (ID_Client) REFERENCES Clienti (ID_Client),  
    FOREIGN KEY (ID_Produs) REFERENCES Produse(ID_Produs)  
);
```

```
INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs,  
DataRecomandare)  
VALUES (1, 1, 1, TO_DATE('2020-11-27','YYYY-MM-DD'));  
  
INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs,  
DataRecomandare)  
VALUES (2, 2, 2, TO_DATE('2020-11-27','YYYY-MM-DD'));  
  
INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs,  
DataRecomandare)  
VALUES (3, 3, 3, TO_DATE('2020-11-27','YYYY-MM-DD'));  
  
INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs,  
DataRecomandare)  
VALUES (4, 4, 4, TO_DATE('2020-11-27','YYYY-MM-DD'));  
  
INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs,  
DataRecomandare)  
VALUES (5, 5, 5, TO_DATE('2020-11-27','YYYY-MM-DD'));  
  
INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs,  
DataRecomandare)  
VALUES (6, 2, 6, TO_DATE('2020-11-27','YYYY-MM-DD'));  
  
INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs,  
DataRecomandare)  
VALUES (7, 3, 7, TO_DATE('2020-11-27','YYYY-MM-DD'));  
  
INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs,  
DataRecomandare)  
VALUES (8, 4, 1, TO_DATE('2020-11-27','YYYY-MM-DD'));
```

```

INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs,
DataRecomandare)
VALUES (9, 5, 2, TO_DATE('2020-11-27','YYYY-MM-DD'));

INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs,
DataRecomandare)
VALUES (10, 1, 3, TO_DATE('2020-11-27','YYYY-MM-DD'));

```

```

INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs, DataRecomandare)
VALUES (1, 1, 1, TO_DATE('2020-11-27','YYYY-MM-DD'));
INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs, DataRecomandare)
VALUES (2, 2, 2, TO_DATE('2020-11-27','YYYY-MM-DD'));
INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs, DataRecomandare)
VALUES (3, 3, 3, TO_DATE('2020-11-27','YYYY-MM-DD'));

INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs, DataRecomandare)
VALUES (4, 4, 4, TO_DATE('2020-11-27','YYYY-MM-DD'));

INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs, DataRecomandare)
VALUES (5, 5, 5, TO_DATE('2020-11-27','YYYY-MM-DD'));

INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs, DataRecomandare)
VALUES (6, 2, 6, TO_DATE('2020-11-27','YYYY-MM-DD'));

INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs, DataRecomandare)
VALUES (7, 3, 7, TO_DATE('2020-11-27','YYYY-MM-DD'));

INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs, DataRecomandare)
VALUES (8, 4, 1, TO_DATE('2020-11-27','YYYY-MM-DD'));

INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs, DataRecomandare)
VALUES (9, 5, 2, TO_DATE('2020-11-27','YYYY-MM-DD'));

INSERT INTO RECOMANDARE(ID_Recomandare, ID_Client, ID_Produs, DataRecomandare)
VALUES (10, 1, 3, TO_DATE('2020-11-27','YYYY-MM-DD'));

```

	ID_RECOMANDARE	ID_CLIENT	ID_PRODUS	DATA RECOMANDARE
1	1	1	1	2020-11-27
2	2	2	2	2020-11-27
3	3	3	3	2020-11-27
4	4	4	4	2020-11-27
5	5	5	5	2020-11-27
6	6	2	6	2020-11-27
7	7	3	7	2020-11-27
8	8	4	1	2020-11-27
9	9	5	2	2020-11-27
10	10	1	3	2020-11-27

- Tabelul Marketing:

```
CREATE TABLE Marketing (  
    ID_Echipa INT PRIMARY KEY,  
    ID_Strategie INT,  
    RataReusita DECIMAL(5, 2),  
    CostMediuPromovare DECIMAL(10, 2)  
);
```

```
CREATE TABLE Marketing (  
    ID_Echipa INT PRIMARY KEY,  
    ID_Strategie INT,  
    RataReusita DECIMAL(5, 2),  
    CostMediuPromovare DECIMAL(10, 2)  
);
```

```
INSERT INTO Marketing (ID_Echipa, ID_STRATEGIE, RataReusita,  
CostMediuPromovare)  
VALUES (1, 1, 0.9, 500.00);  
  
INSERT INTO Marketing (ID_Echipa, ID_STRATEGIE, RataReusita,  
CostMediuPromovare)  
VALUES (2, 2, 0.8, 600.00);  
  
INSERT INTO Marketing (ID_Echipa, ID_STRATEGIE, RataReusita,  
CostMediuPromovare)  
VALUES (3, 3, 0.7, 700.00);  
  
INSERT INTO Marketing (ID_Echipa, ID_STRATEGIE, RataReusita,  
CostMediuPromovare)  
VALUES (4, 4, 0.6, 800.00);  
  
INSERT INTO Marketing (ID_Echipa, ID_STRATEGIE, RataReusita,  
CostMediuPromovare)  
VALUES (5, 5, 0.5, 900.00);
```

```
INSERT INTO Marketing (ID_Echipa, ID_STRATEGIE, RataReusita, CostMediuPromovare)
VALUES (1, 1, 0.9, 500.00);

INSERT INTO Marketing (ID_Echipa, ID_STRATEGIE, RataReusita, CostMediuPromovare)
VALUES (2, 2, 0.8, 600.00);

INSERT INTO Marketing (ID_Echipa, ID_STRATEGIE, RataReusita, CostMediuPromovare)
VALUES (3, 3, 0.7, 700.00);

INSERT INTO Marketing (ID_Echipa, ID_STRATEGIE, RataReusita, CostMediuPromovare)
VALUES (4, 4, 0.6, 800.00);

INSERT INTO Marketing (ID_Echipa, ID_STRATEGIE, RataReusita, CostMediuPromovare)
VALUES (5, 5, 0.5, 900.00);
```

	ID_ECHIPA	ID_STRATEGIE	RATAREUSITA	COSTMEDIUPROMOVARE
1	1	1	0.90	500.00
2	2	2	0.80	600.00
3	3	3	0.70	700.00
4	4	4	0.60	800.00
5	5	5	0.50	900.00

- Tabelul ParteneriExterni:

```
CREATE TABLE ParteneriExterni (  
    ID_Partener INT PRIMARY KEY,  
    NumePartener VARCHAR(100),  
    Adresa VARCHAR(100),  
    Contact VARCHAR(50),  
    ID_Echipa INT,  
    ID_ComandaInsemnata INT,  
    FOREIGN KEY (ID_Echipa) REFERENCES Marketing (ID_Echipa),  
    FOREIGN KEY (ID_ComandaInsemnata) REFERENCES ComenziInsemnate  
    (ID_ComandaInsemnata)  
);
```

```
CREATE TABLE ParteneriExterni (  
    ID_Partener INT PRIMARY KEY,  
    NumePartener VARCHAR(100),  
    Adresa VARCHAR(100),  
    Contact VARCHAR(50),  
    ID_Echipa INT,  
    ID_ComandaInsemnata INT,  
    FOREIGN KEY (ID_Echipa) REFERENCES Marketing (ID_Echipa),  
    FOREIGN KEY (ID_ComandaInsemnata) REFERENCES ComenziInsemnate (ID_ComandaInsemnata)  
);
```

```
INSERT INTO ParteneriExterni (ID_Partener, NumePartener, Adresa, Contact,  
ID_Echipa, ID_ComandaInsemnata)  
VALUES (1, 'ABC Company', 'Str. Principală 123, Orașul X', 'John Doe', 1,  
100);  
  
INSERT INTO ParteneriExterni (ID_Partener, NumePartener, Adresa, Contact,  
ID_Echipa, ID_ComandaInsemnata)  
VALUES (2, 'XYZ Corporation', 'Str. Secundară 456, Orașul Y', 'Jane Smith',  
2, 101);  
  
INSERT INTO ParteneriExterni (ID_Partener, NumePartener, Adresa, Contact,  
ID_Echipa, ID_ComandaInsemnata)  
VALUES (3, 'DEF Industries', 'Str. Principală 789, Orașul Z', 'David  
Johnson', 1, 102);  
  
INSERT INTO ParteneriExterni (ID_Partener, NumePartener, Adresa, Contact,  
ID_Echipa, ID_ComandaInsemnata)  
VALUES (4, 'DEF Industries', 'Str. Principală 789, Orașul Z', 'David  
Johnson', 2, 103);  
  
INSERT INTO ParteneriExterni (ID_Partener, NumePartener, Adresa, Contact,  
ID_Echipa, ID_ComandaInsemnata)
```

```
VALUES (5, 'OMS Industries', 'Str. Principală 229, Orașul Y', 'Kim Mary', 1, 104);
```

```
INSERT INTO ParteneriExterni (ID_Partener, NumePartener, Adresa, Contact, ID_Echipa, ID_ComandaInsemnata)
VALUES (1, 'ABC Company', 'Str. Principală 123, Orașul X', 'John Doe', 1, 100);

INSERT INTO ParteneriExterni (ID_Partener, NumePartener, Adresa, Contact, ID_Echipa, ID_ComandaInsemnata)
VALUES (2, 'XYZ Corporation', 'Str. Secundară 456, Orașul Y', 'Jane Smith', 2, 101);

INSERT INTO ParteneriExterni (ID_Partener, NumePartener, Adresa, Contact, ID_Echipa, ID_ComandaInsemnata)
VALUES (3, 'DEF Industries', 'Str. Principală 789, Orașul Z', 'David Johnson', 1, 102);

INSERT INTO ParteneriExterni (ID_Partener, NumePartener, Adresa, Contact, ID_Echipa, ID_ComandaInsemnata)
VALUES (4, 'DEF Industries', 'Str. Principală 789, Orașul Z', 'David Johnson', 2, 103);

INSERT INTO ParteneriExterni (ID_Partener, NumePartener, Adresa, Contact, ID_Echipa, ID_ComandaInsemnata)
VALUES (5, 'OMS Industries', 'Str. Principală 229, Orașul Y', 'Kim Mary', 1, 104);
```

ID_PARTENER	NUMEPARTENER	ADRESA	CONTACT	ID_ECHIPA	ID_COMANDAINSEMNATA
1	1 ABC Company	Str. Principală 123, Orașul X	John Doe	1	100
2	2 XYZ Corporation	Str. Secundară 456, Orașul Y	Jane Smith	2	101
3	3 DEF Industries	Str. Principală 789, Orașul Z	David Johnson	1	102
4	4 DEF Industries	Str. Principală 789, Orașul Z	David Johnson	2	103
5	5 OMS Industries	Str. Principală 229, Orașul Y	Kim Mary	1	104

- Tabelul ComenziInsemnate :

```
CREATE TABLE ComenziInsemnate (  
    ID_ComandaInsemnata INT PRIMARY KEY,  
    ID_Comanda INT,  
    MotivInsemnare VARCHAR(255),  
    ID_Partener INT,  
    FOREIGN KEY (ID_Comanda) REFERENCES Comenzi (ID_Comanda)  
);
```

```
CREATE TABLE ComenziInsemnate (  
    ID_ComandaInsemnata INT PRIMARY KEY,  
    ID_Comanda INT,  
    MotivInsemnare VARCHAR(255),  
    ID_Partener INT,  
    FOREIGN KEY (ID_Comanda) REFERENCES Comenzi (ID_Comanda)  
);
```

```
INSERT INTO COMENZIINSEMNATE (ID_ComandaInsemnata, ID_Comanda,  
MotivInsemnare, ID_Partener)  
VALUES (100, 1, 'Nu a fost livrata la timp', 1);  
INSERT INTO COMENZIINSEMNATE (ID_ComandaInsemnata, ID_Comanda,  
MotivInsemnare, ID_Partener)  
VALUES (101, 2, 'Colet fragil', 2);  
INSERT INTO COMENZIINSEMNATE (ID_ComandaInsemnata, ID_Comanda,  
MotivInsemnare, ID_Partener)  
VALUES (102, 3, 'Colet deteriorat', 3);  
  
INSERT INTO COMENZIINSEMNATE (ID_ComandaInsemnata, ID_Comanda,  
MotivInsemnare, ID_Partener)  
VALUES (103, 4, 'Colet deteriorat', 4);  
  
INSERT INTO COMENZIINSEMNATE (ID_ComandaInsemnata, ID_Comanda,  
MotivInsemnare, ID_Partener)  
VALUES (104, 5, 'Nu a fost livrata la timp', 5);  
  
INSERT INTO COMENZIINSEMNATE (ID_ComandaInsemnata, ID_Comanda,  
MotivInsemnare, ID_Partener)  
VALUES (105, 2, 'Colet deteriorat', 1);  
  
INSERT INTO COMENZIINSEMNATE (ID_ComandaInsemnata, ID_Comanda,  
MotivInsemnare, ID_Partener)  
VALUES (106, 1, 'Colet fragil', 2);  
  
INSERT INTO COMENZIINSEMNATE (ID_ComandaInsemnata, ID_Comanda,  
MotivInsemnare, ID_Partener)  
VALUES (107, 4, 'Colet deteriorat', 3);  
  
INSERT INTO COMENZIINSEMNATE (ID_ComandaInsemnata, ID_Comanda,  
MotivInsemnare, ID_Partener)  
VALUES (108, 3, 'Nu a fost livrata la timp', 4);
```

```
INSERT INTO COMENZIINSEMNAE (ID_ComandaInsemnata, ID_Comanda,
MotivInsemnare, ID_Partener)
VALUES (109, 5, 'Colet fragil', 5);
```

```
INSERT INTO COMENZIINSEMNAE (ID_ComandaInsemnata, ID_Comanda, MotivInsemnare, ID_Partener)
VALUES (100, 1, 'Nu a fost livrata la timp', 1);
INSERT INTO COMENZIINSEMNAE (ID_ComandaInsemnata, ID_Comanda, MotivInsemnare, ID_Partener)
VALUES (101, 2, 'Colet fragil', 2);
INSERT INTO COMENZIINSEMNAE (ID_ComandaInsemnata, ID_Comanda, MotivInsemnare, ID_Partener)
VALUES (102, 3, 'Colet deteriorat', 3);

INSERT INTO COMENZIINSEMNAE (ID_ComandaInsemnata, ID_Comanda, MotivInsemnare, ID_Partener)
VALUES (103, 4, 'Colet deteriorat', 4);

INSERT INTO COMENZIINSEMNAE (ID_ComandaInsemnata, ID_Comanda, MotivInsemnare, ID_Partener)
VALUES (104, 5, 'Nu a fost livrata la timp', 5);

INSERT INTO COMENZIINSEMNAE (ID_ComandaInsemnata, ID_Comanda, MotivInsemnare, ID_Partener)
VALUES (105, 2, 'Colet deteriorat', 1);

INSERT INTO COMENZIINSEMNAE (ID_ComandaInsemnata, ID_Comanda, MotivInsemnare, ID_Partener)
VALUES (106, 1, 'Colet fragil', 2);

INSERT INTO COMENZIINSEMNAE (ID_ComandaInsemnata, ID_Comanda, MotivInsemnare, ID_Partener)
VALUES (107, 4, 'Colet deteriorat', 3);

INSERT INTO COMENZIINSEMNAE (ID_ComandaInsemnata, ID_Comanda, MotivInsemnare, ID_Partener)
VALUES (108, 3, 'Nu a fost livrata la timp', 4);

INSERT INTO COMENZIINSEMNAE (ID_ComandaInsemnata, ID_Comanda, MotivInsemnare, ID_Partener)
VALUES (109, 5, 'Colet fragil', 5);
```

	ID_COMANDAINSEMNAE	ID_COMANDA	MOTIVINSEMNAE	ID_PARTENER
1	100	1	Nu a fost livrata la timp	1
2	101	2	Colet fragil	2
3	102	3	Colet deteriorat	3
4	104	5	Nu a fost livrata la timp	5
5	103	4	Colet deteriorat	4
6	105	2	Colet deteriorat	1
7	106	1	Colet fragil	2
8	107	4	Colet deteriorat	3
9	108	3	Nu a fost livrata la timp	4
10	109	5	Colet fragil	5

- Tabelul Evenimente :

```
CREATE TABLE Eveniment (
  ID_Eveniment INT PRIMARY KEY,
  NumeEveniment VARCHAR(100),
  Descriere VARCHAR(255),
  DataInceput DATE,
  DataSfarsit DATE,
  ProcentReducere DECIMAL(5, 2),
  ID_Echipa INT,
  FOREIGN KEY (ID_Echipa) REFERENCES Marketing (ID_Echipa)
);
```

```
CREATE TABLE Eveniment (
  ID_Eveniment INT PRIMARY KEY,
  NumeEveniment VARCHAR(100),
  Descriere VARCHAR(255),
  DataInceput DATE,
  DataSfarsit DATE,
  ProcentReducere DECIMAL(5, 2),
  ID_Echipa INT,
  FOREIGN KEY (ID_Echipa) REFERENCES Marketing (ID_Echipa)
);
```

```
INSERT INTO Eveniment(ID_Eveniment, NumeEveniment, Descriere, DataInceput,
DataSfarsit, ProcentReducere, ID_Echipa)
VALUES (1, 'Black Friday', 'Reduceri de 50% la toate produsele',
TO_DATE('2020-11-27','YYYY-MM-DD'), TO_DATE('2020-11-29','YYYY-MM-DD'), 50,
1);

INSERT INTO Eveniment(ID_Eveniment, NumeEveniment, Descriere, DataInceput,
DataSfarsit, ProcentReducere, ID_Echipa)
VALUES (2, 'Craciun', 'Reduceri de 30% la toate produsele', TO_DATE('2020-12-
24','YYYY-MM-DD'), TO_DATE('2020-12-26','YYYY-MM-DD'), 30, 2);

INSERT INTO Eveniment(ID_Eveniment, NumeEveniment, Descriere, DataInceput,
DataSfarsit, ProcentReducere, ID_Echipa)
VALUES (3, 'Revelion', 'Reduceri de 20% la toate produsele', TO_DATE('2020-
12-31','YYYY-MM-DD'), TO_DATE('2021-01-02','YYYY-MM-DD'), 20, 3);

INSERT INTO Eveniment(ID_Eveniment, NumeEveniment, Descriere, DataInceput,
DataSfarsit, ProcentReducere, ID_Echipa)
VALUES (4, 'Valentines Day', 'Reduceri de 40% la toate produsele',
TO_DATE('2021-02-14','YYYY-MM-DD'), TO_DATE('2021-02-16','YYYY-MM-DD'), 40,
4);

INSERT INTO Eveniment(ID_Eveniment, NumeEveniment, Descriere, DataInceput,
DataSfarsit, ProcentReducere, ID_Echipa)
VALUES (5, 'Black Friday', 'Reduceri de 50% la toate produsele',
```

```
TO_DATE('2021-11-26','YYYY-MM-DD'), TO_DATE('2021-11-28','YYYY-MM-DD'), 50,
5);
```

```
INSERT INTO Eveniment(ID_Eveniment, NumeEveniment, Descriere, DataInceput, DataSfarsit, ProcentReducere, ID_Echipa)
VALUES (1, 'Black Friday', 'Reduceri de 50% la toate produsele', TO_DATE('2020-11-27','YYYY-MM-DD'), TO_DATE('2020-11-29','YYYY-MM-DD'), 50, 1);

INSERT INTO Eveniment(ID_Eveniment, NumeEveniment, Descriere, DataInceput, DataSfarsit, ProcentReducere, ID_Echipa)
VALUES (2, 'Craciun', 'Reduceri de 30% la toate produsele', TO_DATE('2020-12-24','YYYY-MM-DD'), TO_DATE('2020-12-26','YYYY-MM-DD'), 30, 2);

INSERT INTO Eveniment(ID_Eveniment, NumeEveniment, Descriere, DataInceput, DataSfarsit, ProcentReducere, ID_Echipa)
VALUES (3, 'Revelion', 'Reduceri de 20% la toate produsele', TO_DATE('2020-12-31','YYYY-MM-DD'), TO_DATE('2021-01-02','YYYY-MM-DD'), 20, 3);

INSERT INTO Eveniment(ID_Eveniment, NumeEveniment, Descriere, DataInceput, DataSfarsit, ProcentReducere, ID_Echipa)
VALUES (4, 'Valentines Day', 'Reduceri de 40% la toate produsele', TO_DATE('2021-02-14','YYYY-MM-DD'), TO_DATE('2021-02-16','YYYY-MM-DD'), 40, 4);

INSERT INTO Eveniment(ID_Eveniment, NumeEveniment, Descriere, DataInceput, DataSfarsit, ProcentReducere, ID_Echipa)
VALUES (5, 'Black Friday', 'Reduceri de 50% la toate produsele', TO_DATE('2021-11-26','YYYY-MM-DD'), TO_DATE('2021-11-28','YYYY-MM-DD'), 50, 5);
```

ID_EVENIMENT	NUMEEVENIMENT	DESCRIERE	DATAINCEPUT	DATASFARSIT	PROCENTREDUCERE	ID_ECHIPA
1	Black Friday	Reduceri de 50% la toate produsele	2020-11-27	2020-11-29	50.00	1
2	Craciun	Reduceri de 30% la toate produsele	2020-12-24	2020-12-26	30.00	2
3	Revelion	Reduceri de 20% la toate produsele	2020-12-31	2021-01-02	20.00	3
4	Valentines Day	Reduceri de 40% la toate produsele	2021-02-14	2021-02-16	40.00	4
5	Black Friday	Reduceri de 50% la toate produsele	2021-11-26	2021-11-28	50.00	5

- Tabelul Promovare:

```
CREATE TABLE Promovare (  
  ID_Promovare INT PRIMARY KEY,  
  ID_Produs INT,  
  DataPromovare DATE,  
  ID_Echipa INT,  
  FOREIGN KEY (ID_Echipa) REFERENCES Marketing (ID_Echipa)  
);
```

```
CREATE TABLE Promovare (  
  ID_Promovare INT PRIMARY KEY,  
  ID_Produs INT,  
  DataPromovare DATE,  
  ID_Echipa INT,  
  FOREIGN KEY (ID_Echipa) REFERENCES Marketing (ID_Echipa)  
);
```

```
INSERT INTO Promovare(ID_PROMOVARE, ID_PRODUS, DATAPROMOVARE, ID_ECHIPA)  
VALUES (1, 1, TO_DATE('2020-11-27','YYYY-MM-DD'), 1);  
  
INSERT INTO Promovare(ID_PROMOVARE, ID_PRODUS, DATAPROMOVARE, ID_ECHIPA)  
VALUES (2, 2, TO_DATE('2020-11-27','YYYY-MM-DD'), 2);  
  
INSERT INTO Promovare(ID_PROMOVARE, ID_PRODUS, DATAPROMOVARE, ID_ECHIPA)  
VALUES (3, 3, TO_DATE('2020-11-27','YYYY-MM-DD'), 3);  
  
INSERT INTO Promovare(ID_PROMOVARE, ID_PRODUS, DATAPROMOVARE, ID_ECHIPA)  
VALUES (4, 4, TO_DATE('2020-11-27','YYYY-MM-DD'), 4);  
  
INSERT INTO Promovare(ID_PROMOVARE, ID_PRODUS, DATAPROMOVARE, ID_ECHIPA)  
VALUES (5, 5, TO_DATE('2020-11-27','YYYY-MM-DD'), 5);
```

```
INSERT INTO Promovare(ID_PROMOVARE, ID_PRODUS, DATAPROMOVARE, ID_ECHIPA)
VALUES (1, 1, TO_DATE('2020-11-27', 'YYYY-MM-DD'), 1);

INSERT INTO Promovare(ID_PROMOVARE, ID_PRODUS, DATAPROMOVARE, ID_ECHIPA)
VALUES (2, 2, TO_DATE('2020-11-27', 'YYYY-MM-DD'), 2);

INSERT INTO Promovare(ID_PROMOVARE, ID_PRODUS, DATAPROMOVARE, ID_ECHIPA)
VALUES (3, 3, TO_DATE('2020-11-27', 'YYYY-MM-DD'), 3);

INSERT INTO Promovare(ID_PROMOVARE, ID_PRODUS, DATAPROMOVARE, ID_ECHIPA)
VALUES (4, 4, TO_DATE('2020-11-27', 'YYYY-MM-DD'), 4);

INSERT INTO Promovare(ID_PROMOVARE, ID_PRODUS, DATAPROMOVARE, ID_ECHIPA)
VALUES (5, 5, TO_DATE('2020-11-27', 'YYYY-MM-DD'), 5);
```

	ID_PROMOVARE	ID_PRODUS	DATAPROMOVARE	ID_ECHIPA
1	1	1	2020-11-27	1
2	2	2	2020-11-27	2
3	3	3	2020-11-27	3
4	4	4	2020-11-27	4
5	5	5	2020-11-27	5

- Tabelul CercetareDezvoltare :

```
CREATE TABLE CercetareDezvoltare (  
  ID_Reprezentant INT PRIMARY KEY,  
  ID_Director INT,  
  NumarCercetariLunar INT,  
  RataReusita DECIMAL(5, 2)  
);
```

```
CREATE TABLE CercetareDezvoltare (  
  ID_Reprezentant INT PRIMARY KEY,  
  ID_Director INT,  
  NumarCercetariLunar INT,  
  RataReusita DECIMAL(5, 2)  
);
```

```
INSERT INTO CercetareDezvoltare (ID_Reprezentant, ID_Director,  
NumarCercetariLunar, RataReusita)  
VALUES (1, 1, 5, 0.8);  
  
INSERT INTO CercetareDezvoltare (ID_Reprezentant, ID_Director,  
NumarCercetariLunar, RataReusita)  
VALUES (2, 2, 8, 0.7);  
  
INSERT INTO CercetareDezvoltare (ID_Reprezentant, ID_Director,  
NumarCercetariLunar, RataReusita)  
VALUES (3, 3, 12, 0.9);  
  
INSERT INTO CercetareDezvoltare (ID_Reprezentant, ID_Director,  
NumarCercetariLunar, RataReusita)  
VALUES (4, 4, 9, 0.75);  
  
INSERT INTO CercetareDezvoltare (ID_Reprezentant, ID_Director,  
NumarCercetariLunar, RataReusita)  
VALUES (5, 5, 11, 0.85);
```

```
INSERT INTO CercetareDezvoltare (ID_Reprezentant, ID_Director, NumarCercetariLunar, RataReusita)
VALUES (1, 1, 5, 0.8);

INSERT INTO CercetareDezvoltare (ID_Reprezentant, ID_Director, NumarCercetariLunar, RataReusita)
VALUES (2, 2, 8, 0.7);

INSERT INTO CercetareDezvoltare (ID_Reprezentant, ID_Director, NumarCercetariLunar, RataReusita)
VALUES (3, 3, 12, 0.9);

INSERT INTO CercetareDezvoltare (ID_Reprezentant, ID_Director, NumarCercetariLunar, RataReusita)
VALUES (4, 4, 9, 0.75);

INSERT INTO CercetareDezvoltare (ID_Reprezentant, ID_Director, NumarCercetariLunar, RataReusita)
VALUES (5, 5, 11, 0.85);
```

	ID_REPREZENTANT	ID_DIRECTOR	NUMARCERCETARILUNAR	RATAREUSITA
1	1	1	5	0.80
2	2	2	8	0.70
3	3	3	12	0.90
4	4	4	9	0.75
5	5	5	11	0.85

- Tabelul Voluntari :

```
CREATE TABLE Voluntari (  
    ID_Voluntar INT PRIMARY KEY,  
    MedieEsecAlergii DECIMAL(5, 2),  
    MedieFeedback DECIMAL(5, 2),  
    ID_Reprezentant INT,  
    FOREIGN KEY (ID_Reprezentant) REFERENCES CercetareDezvoltare  
    (ID_Reprezentant)  
);
```

```
CREATE TABLE Voluntari (  
    ID_Voluntar INT PRIMARY KEY,  
    MedieEsecAlergii DECIMAL(5, 2),  
    MedieFeedback DECIMAL(5, 2),  
    ID_Reprezentant INT,  
    FOREIGN KEY (ID_Reprezentant) REFERENCES CercetareDezvoltare (ID_Reprezentant)  
);
```

```
INSERT INTO Voluntari (ID_Voluntar, MedieEsecAlergii, MedieFeedback,  
ID_Reprezentant)  
VALUES (1, 0.2, 4.5, 1);  
  
INSERT INTO Voluntari (ID_Voluntar, MedieEsecAlergii, MedieFeedback,  
ID_Reprezentant)  
VALUES (2, 0.5, 4.2, 2);  
  
INSERT INTO Voluntari (ID_Voluntar, MedieEsecAlergii, MedieFeedback,  
ID_Reprezentant)  
VALUES (3, 0.1, 4.8, 1);  
  
INSERT INTO Voluntari (ID_Voluntar, MedieEsecAlergii, MedieFeedback,  
ID_Reprezentant)  
VALUES (4, 0.3, 4.6, 4);  
  
INSERT INTO Voluntari (ID_Voluntar, MedieEsecAlergii, MedieFeedback,  
ID_Reprezentant)  
VALUES (5, 0.4, 4.3, 3);
```

```
INSERT INTO Voluntari (ID_Voluntar, MedieEsecAlergii, MedieFeedback, ID_Reprezentant)
VALUES (1, 0.2, 4.5, 1);

INSERT INTO Voluntari (ID_Voluntar, MedieEsecAlergii, MedieFeedback, ID_Reprezentant)
VALUES (2, 0.5, 4.2, 2);

INSERT INTO Voluntari (ID_Voluntar, MedieEsecAlergii, MedieFeedback, ID_Reprezentant)
VALUES (3, 0.1, 4.8, 1);

INSERT INTO Voluntari (ID_Voluntar, MedieEsecAlergii, MedieFeedback, ID_Reprezentant)
VALUES (4, 0.3, 4.6, 4);

INSERT INTO Voluntari (ID_Voluntar, MedieEsecAlergii, MedieFeedback, ID_Reprezentant)
VALUES (5, 0.4, 4.3, 3);
```

	ID_VOLUNTAR	MEDIEESECALERGII	MEDIEFEEDBACK	ID_REPREZENTANT
1	1	0.20	4.50	1
2	2	0.50	4.20	2
3	3	0.10	4.80	1
4	4	0.30	4.60	4
5	5	0.40	4.30	3

- Tabelul Laboratoare :

```
CREATE TABLE Laboratoare (  
  ID_Laborator INT PRIMARY KEY,  
  NumeLaborator VARCHAR(100),  
  Adresa VARCHAR(100),  
  Contact VARCHAR(50),  
  ID_Reprezentant INT,  
  FOREIGN KEY (ID_Reprezentant) REFERENCES CercetareDezvoltare  
  (ID_Reprezentant)  
);
```

```
CREATE TABLE Laboratoare (  
  ID_Laborator INT PRIMARY KEY,  
  NumeLaborator VARCHAR(100),  
  Adresa VARCHAR(100),  
  Contact VARCHAR(50),  
  ID_Reprezentant INT,  
  FOREIGN KEY (ID_Reprezentant) REFERENCES CercetareDezvoltare (ID_Reprezentant)  
);
```

```
INSERT INTO Laboratoare (ID_Laborator, NumeLaborator, Adresa, Contact,  
ID_Reprezentant)  
VALUES (1, 'Laboratorul 1', 'Str. Principală 123, Orașul X', 'John Doe', 1);  
  
INSERT INTO Laboratoare (ID_Laborator, NumeLaborator, Adresa, Contact,  
ID_Reprezentant)  
VALUES (2, 'Laboratorul 2', 'Str. Secundară 456, Orașul Y', 'Jane Smith', 2);  
  
INSERT INTO Laboratoare (ID_Laborator, NumeLaborator, Adresa, Contact,  
ID_Reprezentant)  
VALUES (3, 'Laboratorul 3', 'Str. Principală 789, Orașul Z', 'David Johnson',  
3);  
  
INSERT INTO Laboratoare (ID_Laborator, NumeLaborator, Adresa, Contact,  
ID_Reprezentant)  
VALUES (4, 'Laboratorul 4', 'Str. Principală 229, Orașul Y', 'Kim Mary', 4);  
  
INSERT INTO Laboratoare (ID_Laborator, NumeLaborator, Adresa, Contact,  
ID_Reprezentant)  
VALUES (5, 'Laboratorul 5', 'Str. Principală 229, Orașul Y', 'Kim Mary', 5);
```

```

INSERT INTO Laboratoare (ID_Laborator, NumeLaborator, Adresa, Contact, ID_Reprezentant)
VALUES (1, 'Laboratorul 1', 'Str. Principală 123, Orașul X', 'John Doe', 1);

INSERT INTO Laboratoare (ID_Laborator, NumeLaborator, Adresa, Contact, ID_Reprezentant)
VALUES (2, 'Laboratorul 2', 'Str. Secundară 456, Orașul Y', 'Jane Smith', 2);

INSERT INTO Laboratoare (ID_Laborator, NumeLaborator, Adresa, Contact, ID_Reprezentant)
VALUES (3, 'Laboratorul 3', 'Str. Principală 789, Orașul Z', 'David Johnson', 3);

INSERT INTO Laboratoare (ID_Laborator, NumeLaborator, Adresa, Contact, ID_Reprezentant)
VALUES (4, 'Laboratorul 4', 'Str. Principală 229, Orașul Y', 'Kim Mary', 4);

INSERT INTO Laboratoare (ID_Laborator, NumeLaborator, Adresa, Contact, ID_Reprezentant)
VALUES (5, 'Laboratorul 5', 'Str. Principală 229, Orașul Y', 'Kim Mary', 5);

```

	ID_LABORATOR	NUMELABORATOR	ADRESA	CONTACT	ID_REPREZENTANT
1		1 Laboratorul 1	Str. Principală 123, Orașul X	John Doe	1
2		2 Laboratorul 2	Str. Secundară 456, Orașul Y	Jane Smith	2
3		3 Laboratorul 3	Str. Principală 789, Orașul Z	David Johnson	3
4		4 Laboratorul 4	Str. Principală 229, Orașul Y	Kim Mary	4
5		5 Laboratorul 5	Str. Principală 229, Orașul Y	Kim Mary	5

- Tabelul Ingrediente :

```
CREATE TABLE Ingrediente (  
    ID_Ingredient INT PRIMARY KEY,  
    NumeIngredient VARCHAR(100),  
    Descriere VARCHAR(255),  
    ID_Produs INT,  
    FOREIGN KEY (ID_Produs) REFERENCES PRODUSE (ID_Produs)  
);
```

```
CREATE TABLE Ingrediente (  
    ID_Ingredient INT PRIMARY KEY,  
    NumeIngredient VARCHAR(100),  
    Descriere VARCHAR(255),  
    ID_Produs INT,  
    FOREIGN KEY (ID_Produs) REFERENCES PRODUSE (ID_Produs)  
);
```

```
INSERT INTO Ingrediente (ID_Ingredient, NumeIngredient, Descriere, ID_Produs)  
VALUES (1, 'Ulei de argan', 'Ingredient hidratant pentru îngrijirea pielii',  
1);  
  
INSERT INTO Ingrediente (ID_Ingredient, NumeIngredient, Descriere, ID_Produs)  
VALUES (2, 'Extract de aloe vera', 'Ingredient calmant pentru ten sensibil',  
1);  
  
INSERT INTO Ingrediente (ID_Ingredient, NumeIngredient, Descriere, ID_Produs)  
VALUES (3, 'Acid hialuronic', 'Ingredient hidratant și anti-îmbătrânire', 2);  
  
INSERT INTO Ingrediente (ID_Ingredient, NumeIngredient, Descriere, ID_Produs)  
VALUES (4, 'Extract de ceai verde', 'Ingredient calmant pentru ten sensibil',  
2);  
  
INSERT INTO Ingrediente (ID_Ingredient, NumeIngredient, Descriere, ID_Produs)  
VALUES (5, 'Extract de trandafir', 'Ingredient cu proprietăți tonifiante și  
revigorante', 1);
```

```

INSERT INTO Ingrediente (ID_Ingredient, NumeIngredient, Descriere, ID_Produs)
VALUES (1, 'Ulei de argan', 'Ingredient hidratant pentru îngrijirea pielii', 1);

INSERT INTO Ingrediente (ID_Ingredient, NumeIngredient, Descriere, ID_Produs)
VALUES (2, 'Extract de aloe vera', 'Ingredient calmant pentru ten sensibil', 1);

INSERT INTO Ingrediente (ID_Ingredient, NumeIngredient, Descriere, ID_Produs)
VALUES (3, 'Acid hialuronic', 'Ingredient hidratant și anti-îmbătrânire', 2);

INSERT INTO Ingrediente (ID_Ingredient, NumeIngredient, Descriere, ID_Produs)
VALUES (4, 'Extract de ceai verde', 'Ingredient calmant pentru ten sensibil', 2);

INSERT INTO Ingrediente (ID_Ingredient, NumeIngredient, Descriere, ID_Produs)
VALUES (5, 'Extract de trandafir', 'Ingredient cu proprietăți tonifiante și revigorante', 1);

```

ID_INGREDIENT	NUMEINGREDIENT	DESCRIERE	ID_PRODUS
1	1 Ulei de argan	Ingredient hidratant pentru îngrijirea pielii	1
2	2 Extract de aloe vera	Ingredient calmant pentru ten sensibil	1
3	3 Acid hialuronic	Ingredient hidratant și anti-îmbătrânire	2
4	4 Extract de ceai verde	Ingredient calmant pentru ten sensibil	2
5	5 Extract de trandafir	Ingredient cu proprietăți tonifiante și revigorante	1

- Tabelul Stocuri :

```
CREATE TABLE Stocuri (  
    ID_Stoc INT PRIMARY KEY,  
    ID_Produs INT,  
    Cantitate INT,  
    DataActualizare DATE,  
    FOREIGN KEY (ID_Produs) REFERENCES Produse (ID_Produs)  
);
```

```
CREATE TABLE Stocuri (  
    ID_Stoc INT PRIMARY KEY,  
    ID_Produs INT,  
    Cantitate INT,  
    DataActualizare DATE,  
    FOREIGN KEY (ID_Produs) REFERENCES Produse (ID_Produs)  
);
```

```
INSERT INTO Stocuri (ID_Stoc, ID_Produs, Cantitate, DataActualizare)  
VALUES (1, 1, 50, TO_Date('2023-05-01','YYYY-MM-DD'));  
  
INSERT INTO Stocuri (ID_Stoc, ID_Produs, Cantitate, DataActualizare)  
VALUES (2, 2, 50, TO_Date('2023-05-04','YYYY-MM-DD'));  
  
INSERT INTO Stocuri (ID_Stoc, ID_Produs, Cantitate, DataActualizare)  
VALUES (3, 3, 50, TO_Date('2023-05-05','YYYY-MM-DD'));  
  
INSERT INTO Stocuri (ID_Stoc, ID_Produs, Cantitate, DataActualizare)  
VALUES (4, 4, 50, TO_Date('2023-05-03','YYYY-MM-DD'));  
  
INSERT INTO Stocuri (ID_Stoc, ID_Produs, Cantitate, DataActualizare)
```

```
INSERT INTO Stocuri (ID_Stoc, ID_Produs, Cantitate, DataActualizare)
VALUES (1, 1, 50, TO_Date('2023-05-01','YYYY-MM-DD'));

INSERT INTO Stocuri (ID_Stoc, ID_Produs, Cantitate, DataActualizare)
VALUES (2, 2, 50, TO_Date('2023-05-04','YYYY-MM-DD'));

INSERT INTO Stocuri (ID_Stoc, ID_Produs, Cantitate, DataActualizare)
VALUES (3, 3, 50, TO_Date('2023-05-05','YYYY-MM-DD'));

INSERT INTO Stocuri (ID_Stoc, ID_Produs, Cantitate, DataActualizare)
VALUES (4, 4, 50, TO_Date('2023-05-03','YYYY-MM-DD'));

INSERT INTO Stocuri (ID_Stoc, ID_Produs, Cantitate, DataActualizare)
VALUES (5, 5, 50, TO_Date('2023-05-02','YYYY-MM-DD'));
```

	ID_STOC	ID_PRODUS	CANTITATE	DATAACTUALIZARE
1	1	1	50	2023-05-01
2	2	2	50	2023-05-04
3	3	3	50	2023-05-05
4	4	4	50	2023-05-03
5	5	5	50	2023-05-02

CERINTA 12

Formulați în limbaj natural și implementați 5 cereri SQL complexe ce vor utiliza, în ansamblul lor, următoarele elemente:

- 1) • **subcereri sincronizate în care intervin cel puțin 3 tabele**
 - 2) • **subcereri nesincronizate în clauza FROM**
 - 3) • **grupări de date cu subcereri nesincronizate în care intervin cel puțin 3 tabele, funcții grup, filtrare la nivel de grupuri (în cadrul aceleiasi cereri)**
 - 4) • **ordonări și utilizarea funcțiilor NVL și DECODE (în cadrul aceleiasi cereri)**
 - 5) • **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)**

- 1) Aceasta cerere afiseaza numele clientilor, numele produselor si data recomandarilor pentru toate produsele cosmetice recomandate de clienti.

```
SELECT
    C.Nume, P.NumeProdus, R.DataRecomandare
FROM Clienti C
JOIN Recomandare R ON C.ID_Client = R.ID_Client
JOIN Produse P ON R.ID_Produs = P.ID_Produs
WHERE P.Categorie = 'Machiaj';
```

	NUME	NUMEPRODUS	DATARECOMANDARE
1	Popescu	Rimel	2020-11-27
2	Ionescu	Ruj	2020-11-27
3	Pop	Fond de ten	2020-11-27
4	Pop	Rimel	2020-11-27
5	Popescu	Ruj	2020-11-27

Observatie : aceste cerinte au fost create si rulate dupa efectuarea cerintei 13.

- 2) Aceasta cerere returneaza numele, prenumele si numarul de comenzi pentru fiecare client din baza de date.

```
SELECT C.Nume, C.Prenume,
       (SELECT
          COUNT(*) F
        FROM Comenzi
         WHERE ID_Client = C.ID_Client) AS NumarComenzi
FROM Clienti C;
```

	NUME	PRENUME	NUMARCOMENZI
1	Popescu	Ana	1
2	Ionescu	Maria	1
3	Popa	Andrei	1
4	Pop	Andreea	1
5	Popescu	Andrei	1

- 3) Aceasta cerere grupeaza produsele dupa categorii, numara produsele din fiecare categorie, calculeaza pretul mediu de vanzare si returneaza rezultatele doar pentru categoriile cu mai mult de un produs.

```
SELECT
  P.Categorie,
  COUNT(*) AS NumarProduse,
  AVG(P.PretVanzare) AS PretMediu
FROM Produse P
JOIN Comenzi C ON P.ID_Comanda = C.ID_Comanda
JOIN Clienti Cl ON C.ID_Client = Cl.ID_Client
GROUP BY P.Categorie
HAVING COUNT(*) > 1;
```

	CATEGORIE	NUMARPRODUSE	PRETMEDIU
1	Ingrijirea pielii	2	45
2	Machiaj	3	25

- 4) Aceasta cerere selecteaza numele produsului, pretul de vanzare, review-ul (afiseaza "Fara review" daca este nul) si converteste starea comenzii intr-o forma mai simpla pentru afisare, apoi ordoneaza rezultatele dupa pretul de vanzare in ordine descrescatoare.

```
SELECT NumeProdus, PretVanzare,
       NVL(Review, 'Fără review') AS Review,
       DECODE(StareComanda, 'In curs de procesare', 'In procesare',
              'Incheiata') AS StareComandaFinala
FROM Produse P
JOIN Comenzi C ON P.ID_Comanda = C.ID_Comanda
ORDER BY PretVanzare DESC;
```

	CATEGORIE	NUMARPRODUSE	PRETMEDIU
1	Ingrijirea pielii	2	45
2	Machiaj	3	25

- 5) Aceasta cerere combina ambele cerinte, utilizand functii pe siruri de caractere (SUBSTR si LENGTH), functii pe date calendaristice (TO_CHAR si EXTRACT), o expresie CASE si un bloc de cerere (clauza WITH) cu doua blocuri de cerere separate ("ProduseCategorie" si "DataPromovare").

```
WITH ProduseCategorie AS (
  SELECT P.Categorie, COUNT(*) AS NumarProduse
  FROM Produse P
  GROUP BY P.Categorie
), DataPromovare AS (
  SELECT ID_Produs, MAX(DataPromovare) AS UltimaDataPromovare
  FROM Promovare
  GROUP BY ID_Produs
)
SELECT P.NumeProdus,
       SUBSTR(P.NumeProdus, 1, 5) AS Subsirl1,
       LENGTH(P.NumeProdus) AS LungimeNume,
       TO_CHAR(CURRENT_DATE, 'YYYY-MM-DD') AS DataCurenta,
       EXTRACT(YEAR FROM CURRENT_DATE) AS AnCurent,
       CASE WHEN P.PretVanzare > P.PretFabricare THEN 'Profit' ELSE
       'Pierdere' END AS SituatieProfit
FROM Produse P
JOIN ProduseCategorie PC ON P.Categorie = PC.Categorie
JOIN DataPromovare DP ON P.ID_Produs = DP.ID_Produs;
```

	NUMEPRODUS	SUBSIR1	LUNGIMENUME	DATACURRENTA	ANCURENT	SITUATIEPROFIT
1	Crema de fata	Crema	13	2023-05-26	2023	Profit
2	Rimel	Rimel	5	2023-05-26	2023	Profit
3	Ruj	Ruj	3	2023-05-26	2023	Profit
4	Fond de ten	Fond	11	2023-05-26	2023	Profit
5	Gel de curatare	Gel d	16	2023-05-26	2023	Profit

CERINTA 13

Implementarea a 3 operații de actualizare și de ștergere a datelor utilizând subcereri.

1) Actualizarea unei înregistrări folosind o subcerere:

Această cerere actualizează prețul de vânzare al produsului cu cel mai mare ID din categoria "Îngrijirea tenului" la valoarea medie a prețurilor de vânzare ale tuturor produselor din aceeași categorie.

```
UPDATE Produse
SET PretVanzare = (
    SELECT AVG(PretVanzare)
    FROM Produse
    WHERE Categorie = 'Îngrijirea tenului'
)
WHERE ID_Produs = (
    SELECT MAX(ID_Produs)
    FROM Produse
    WHERE Categorie = 'Îngrijirea tenului'
);
```

	NUMEPRODUS	SUBSIR1	LUNGIMENUME	DATAACURENTA	ANCURENT	SITUATIEPROFIT
1	Crema de fata	Crema	13	2023-05-26	2023	Profit
2	Rimel	Rimel	5	2023-05-26	2023	Profit
3	Ruj	Ruj	3	2023-05-26	2023	Profit
4	Fond de ten	Fond	11	2023-05-26	2023	Profit
5	Gel de curatare	Gel d	16	2023-05-26	2023	Profit

2) Actualizarea inregistrarilor folosind o subcerere cu conditie:

Aceasta cerere elimina legatura dintre clienti si comenzile plasate inainte de 1 ianuarie 2022, setand valoarea coloanei "ID_Comanda" la NULL pentru clientii care indeplinesc aceasta conditie.

```
UPDATE Clienti
SET ID_Comanda = NULL
WHERE ID_Client IN (
    SELECT ID_Client
    FROM Comenzi
    WHERE DataComanda < TO_DATE('2022-01-01','YYYY-MM-DD')
);
```

ID_CLIENT	NUME	PRENUME	ADRESA	NRTelefon	EMAIL	ID_COMANDA
1	1 Popescu	Ana	Str. Florilor, Nr. 10	0721122334	ana.popescu@gmail.com	<null>
2	2 Ionescu	Maria	Str. Florilor, Nr. 11	0721122335	ionescu.maria@gmail.com	<null>
3	3 Popa	Andrei	Str. Florilor, Nr. 12	0721122336	popa.andrei@gmail.com	<null>
4	4 Pop	Andreea	Str. Florilor, Nr. 13	0721122337	pop.andreea@gmail.com	<null>
5	5 Popescu	Andrei	Str. Florilor, Nr. 14	0721122338	popescu.andrei@yahoo.com	<null>

3) Stergerea inregistrarilor folosind o subcerere:

Aceasta cerere sterge toate inregistrarile din tabela "Produse" care au un ID_Produs asociat in tabela "Stocuri" si au o cantitate de 0.

```
DELETE FROM Produse
WHERE ID_Produs IN (
    SELECT ID_Produs
    FROM Stocuri
    WHERE Cantitate = 0
);
```

ID_STOC	ID_PRODUS	CANTITATE	DATAACTUALIZARE
1	1	50	2023-05-01
2	2	50	2023-05-04
3	3	50	2023-05-05
4	4	50	2023-05-03
5	5	50	2023-05-02

CERINTA 14

Crearea unei vizualizări complexe. Dați un exemplu de operație LMD permisă pe vizualizarea respectivă și un exemplu de operație LMD nepermisă.

Exemplul de comanda LMD permisa este o interogare SELECT complexa care returneaza categoria produselor si pretul mediu de vanzare pentru fiecare categorie, dar doar pentru categoriile in care pretul mediu depaseste 15

```
SELECT Produse.Categorie, AVG(Produse.PretVanzare) AS PretMediu
FROM Produse
GROUP BY Produse.Categorie
HAVING AVG(Produse.PretVanzare) > 15;
```

```
SELECT Produse.Categorie, AVG(Produse.PretVanzare) AS PretMediu
FROM Produse
GROUP BY Produse.Categorie
HAVING AVG(Produse.PretVanzare) > 15;
```

[illegible]

Exemplul de comanda LMD nepermisa este o combinatie de mai multe comenzi care conduc la o modificare nedorita a datelor din baza de date.

```
UPDATE Clienti  
SET Nume = 'Smith', NrTelefon = '1234567890';
```

```
UPDATE Clienti  
SET Nume = 'Smith', NrTelefon = '1234567890';
```

Aceasta comanda ar putea actualiza numele si numarul de telefon pentru toate inregistrarile din tabela Clienti, fara a avea o clauza de restrictive (cum ar fi WHERE). Acest lucru ar duce la inlocuirea datelor existente cu valorile specificate in comanda pentru toate inregistrarile, ceea ce ar putea fi o modificare majora si ireversibila a informatiilor despre client

CERINTA 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.

- Cerere ce utilizeaza operatia outer-join pe minimum 4 tabele

```
SELECT C.ID_Comanda, C.DataComanda, P.NumeProdus, R.ID_Recomandare
FROM Comenzi C
LEFT JOIN Produse P ON C.ID_Comanda = P.ID_Comanda
LEFT JOIN ParteneriExterni PE ON C.ID_Comanda = PE.ID_ComandaInsemnata
LEFT JOIN RECOMANDARE R ON C.ID_Client = R.ID_Client;
```

```
SELECT C.ID_Comanda, C.DataComanda, P.NumeProdus, R.ID_Recomandare
FROM Comenzi C
LEFT JOIN Produse P ON C.ID_Comanda = P.ID_Comanda
LEFT JOIN ParteneriExterni PE ON C.ID_Comanda = PE.ID_ComandaInsemnata
LEFT JOIN RECOMANDARE R ON C.ID_Client = R.ID_Client;
```

	ID_COMANDA	DATACOMMANDA	NUMEPRODUS	ID_RECOMANDARE
1	1	2021-05-25	Crema hidratanta	1
2	1	2021-05-25	Rimel waterproof	1
3	2	2021-05-25	Ruj mat	2
4	3	2021-05-26	Parfum floral	3
5	3	2021-05-26	Lotiune tonica	3
6	4	2021-05-27	Gel de dus	4
7	5	2021-05-27	Fond de ten	5
8	2	2021-05-25	Ruj mat	6
9	3	2021-05-26	Parfum floral	7
10	3	2021-05-26	Lotiune tonica	7
11	4	2021-05-27	Gel de dus	8
12	5	2021-05-27	Fond de ten	9
13	1	2021-05-25	Crema hidratanta	10
14	1	2021-05-25	Rimel waterproof	10

Rezultatul interogării va conține coloanele specificate în SELECT și va afișa înregistrări care corespund condițiilor de join definite între tabele. Folosind LEFT JOIN, se vor afișa toate înregistrările din tabela "Comenzi" și se vor adăuga informațiile corespunzătoare din celelalte tabele, dacă există potriviri.

- Cerere ce utilizeaza operatia division

```
SELECT
    C.Nume, C.Prenume
FROM Clienti C
WHERE NOT EXISTS (
    SELECT *
    FROM Comenzi
    WHERE ID_Client = C.ID_Client
    AND DataComanda > TO_DATE('2022-01-01', 'YYYY-MM-DD')
);
```

```
--Exemplu de cerere cu operatia DIVISION:
SELECT
    C.Nume, C.Prenume
FROM Clienti C
WHERE NOT EXISTS (
    SELECT *
    FROM Comenzi
    WHERE ID_Client = C.ID_Client
    AND DataComanda > TO_DATE('2022-01-01', 'YYYY-MM-DD')
);
```

	NUME	PRENUME
1	Popescu	Ana
2	Ionescu	Maria
3	Popa	Andrei
4	Pop	Andreea
5	Popescu	Andrei

Rezultatul interogării va conține numele și prenumele clienților care nu au înregistrări în tabela "Comenzi" cu o dată de comandă ulterioară datei de 1 ianuarie 2022.

- Cerere care implementeaza analiza top-n

```
SELECT
  *
FROM (
  SELECT
    C.Nume, C.Prenume, COUNT(*) AS NumarComenzi
  FROM Clienti C
  JOIN Comenzi CO ON C.ID_Client = CO.ID_Client
  GROUP BY C.Nume, C.Prenume
  ORDER BY COUNT(*) DESC
)
WHERE ROWNUM <= 3;
```

```
--Exemplu de cerere pentru analiza top-N:
SELECT
  *
FROM (
  SELECT
    C.Nume, C.Prenume, COUNT(*) AS NumarComenzi
  FROM Clienti C
  JOIN Comenzi CO ON C.ID_Client = CO.ID_Client
  GROUP BY C.Nume, C.Prenume
  ORDER BY COUNT(*) DESC
)
WHERE ROWNUM <= 3;
```

	NUME	PRENUME	NUMARCOMENZI
1	Popescu	Ana	1
2	Pop	Andreea	1
3	Popescu	Andrei	1

Rezultatul final al interogarii va contine toate coloanele (inclusive nume, prenume si numarul de comenzi) pentru primele 3 inregistrari obtinute din interogarea interna, care returneaza clientii cu cele mai multe comenzi, ordonate descendent dupa numarul de comenzi