

Gestiune distribuitor produse cosmetice

Dimoiu Mihai – Lorin

Etapa de proiectare:

Etapa proiectarii a constat in culegerea informatiilor necesare realizarii proiectului intitulat “Gestiune distribuitor produse cosmetice” din diverse surse informative precum mediul online si spatiul mediatic dedicat acestui domeniu. Odata ajuns in posesia acestora, a urmat procesul de creare a unei concordante si constituirea unor schite atat pentru pentru un concept de baza de date, cat si pentru clasele utilizate bazate pe codul sursa culminand cu o schita pentru realizarea interdependentiei claselor cu interfata.

De asemenea, infaptuirea aplicatiilor de test a fost un urmator pas important pentru testarea fiecărei functionalitati din aplicatie, urmand ca acestea din urma sa fie implementate intr-o aplicatie unitara.

Am ales Microsoft SQL Server 2016 ca si sistem de gestiune a bazelor de date si scrierea codului aplicatiei a fost facut in Java cu ajutorul IDE-ului IntelliJ IDEA, iar pentru interfata am utilizat programul JavaFX Scene Builder deoarece am constat ca a facilitat crearea acesteia avand un raport timp-efort foarte bun.

Descrierea tabelelor:

Tabelele pe care le-am creat in baza de date sunt urmatoarele:

- **Produse_Comenzi** : Acest tabel este cel de legatura pentru relatia M:M intre “Produse”, respectiv “Comenzi”
- **Comenzi** : In acest tabel sunt stocate date despre data unei comenzi, dar si starea acesteia
- **Produse** : Acest tabel inglobeaza date despre numele produselor, cantitatea lor (in mL), pret, etc.
- **Producatori** : Tabelul este creat pentru a evidentia producatorii
- **Categorii**: Aici se afla categoriile pentru fiecare produs in parte
- **Client si Adrese**: Sunt inregistrate date despre client precum numele si adresa de livrare

Relatiile dintre tabele:

Relatia M:M se realizeaza prin tabela de legatura Produse_Comenzi avand ca si cheie primara combinatia de coloane (ID_Comanda, ID_Produs) regasindu-se aceste campuri si in tabelele Produse, respectiv, Comenzi.

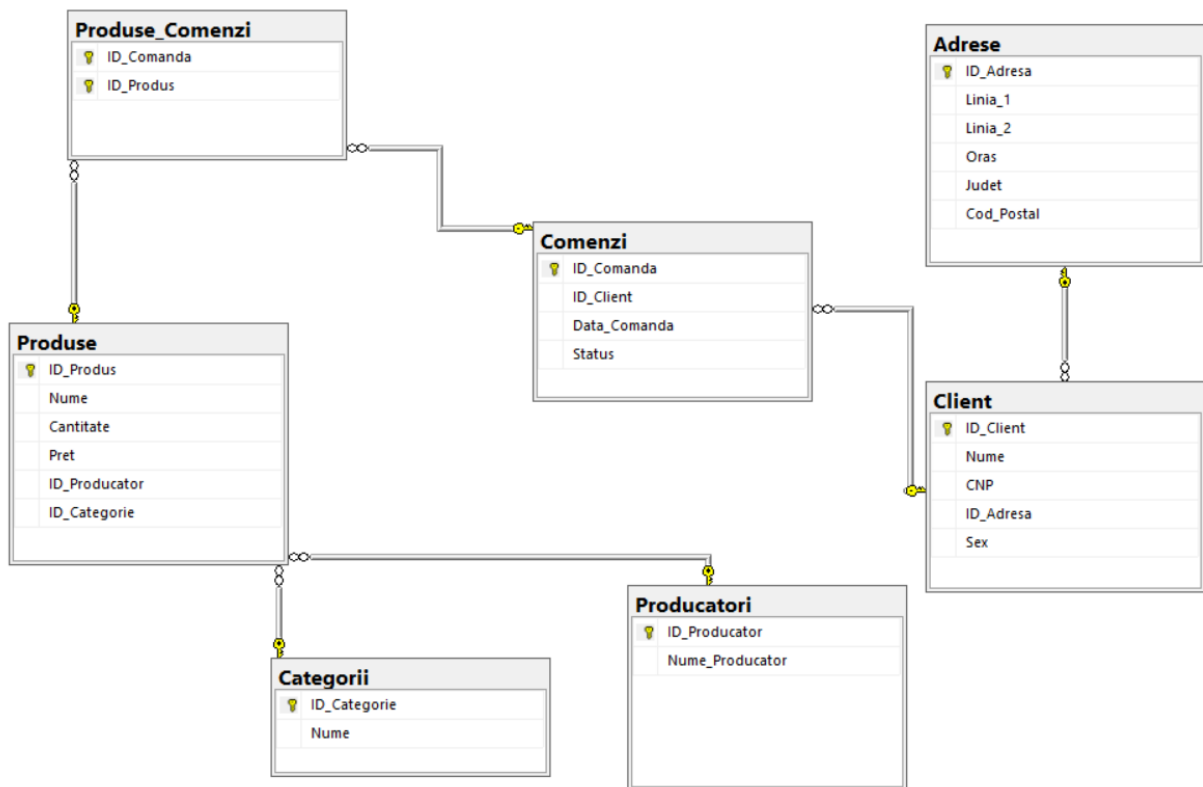
Relatia 1:M Produse – Categori, respectiv Produse – Producatori (pentru a simplifica, am ales ca produsele sa apartina doar unei categorii, respectiv un produs unui singur producator)

Relatia 1:1 Client – Adrese

Relatia 1:M Client – Comenzi

Constrangeri :

- 1) Din motive rationale pentru inlesinirea proiectarii bazei de date am ales sa pun toate campurile din toate tabelele sa nu accepte valoare NULL.
- 2) CNP-ul unui client sa accepte doar 13 cifre.
- 3) Sexul unui client sa fie 'F' (Feminin) daca s-a omis la introducerea in baza de date.
De asemenea, sa nu accepte altceva inafara de 'F' sau 'M'.



Interogari:

1. Selectarea categoriilor pe baza introducerii de la tastatura a unui nume de categorie.

```

SELECT [Proiect].[dbo].Categori.Nume,
[Proiect].[dbo].Produse.Nume,[Proiect].[dbo].Produse.Pret
FROM [Proiect].[dbo].Categori INNER JOIN
[Proiect].[dbo].Produse ON Categori.ID_Categorie =
[Proiect].[dbo].Produse.ID_Categorie
WHERE (lower([Proiect].[dbo].Categori.Nume)LIKE '%' + field1.getText() + '%');
  
```

2. Afisarea tuturor clientilor care au efectuat cumparaturi

```

SELECT cl.ID_Client,cl.Nume, adr.Linia_1 + ' ' + adr.Linia_2
AS 'Adresa',adr.Oras,adr.Judet,adr.Cod_Postal
FROM Proiect.dbo.Client cl INNER JOIN Adrese adr ON cl.ID_Adresa = adr.ID_Adresa
WHERE cl.ID_Client IN(SELECT cl1.ID_Client
FROM Client cl1, Comenzi cm
WHERE cl.ID_Client = cm.ID_Client)
  
```

3. Afisarea tuturor produselor care au fost cumparate pentru ulterioare statistici

```
SELECT pr.Nume,pr.Pret
FROM Proiect.dbo.Produse pr
WHERE pr.ID_Produs IN(SELECT pc.ID_Produs
FROM Proiect.dbo.Client c1 INNER JOIN Proiect.dbo.Comenzi cm ON c1.ID_Client =
cm.ID_Client INNERJOIN Proiect.dbo.Produse_Comenzi pc on cm.ID_Comanda =
pc.ID_Comanda)
```

4. Afisarea clientilor actuali din baza de date

```
SELECT [Nume],[CNP],[Sex],[Linia_1] + ' ' + [Linia_2] AS 'Adresa',
[Oras],[Judet],[Cod_Postal]
FROM [Proiect].[dbo].[Client] INNER JOIN [Proiect].[dbo].[Adrese] ON
[Proiect].[dbo].[Client].ID_Adresa = [Proiect].[dbo].[Adrese].ID_Adresa
```

5. Afisarea comenzilor pe baza numelui introdus din tastatura

```
SELECT c1.Nume,pr.Nume, pr.Pret
FROM Produse_Comenzi pc INNER JOIN Produse pr ON pc.ID_Produs = pr.ID_Produs
INNERJOIN Comenzi cm ON pc.ID_Comanda = cm.ID_Comanda INNERJOIN Client c1 ON
cm.ID_Client = c1.ID_Client
WHERE (lower(c1.Nume)LIKE '%' + field1.getText() + '%')
```

6. Selectarea clientilor care nu au facut cumparaturi pentru a ii adauga pe lista de inactivitate pentru a putea fi eliminati ulterior

```
SELECT c1.ID_Client,c1.Nume, adr.Linia_1 + ' ' + adr.Linia_2
AS 'Adresa',adr.Oras,adr.Judet,adr.Cod_Postal
FROM Proiect.dbo.Client c1 INNER JOIN Adrese adr ON c1.ID_Adresa = adr.ID_Adresa
WHERE c1.ID_Client NOTIN(SELECT c11.ID_Client
FROM Client c11, Comenzi cm
WHERE c1.ID_Client = cm.ID_Client)
```

7. Selectarea produselor cu criteriul introdus din tastatura

```
SELECT [Nume],[Cantitate],[Pret],[Producatori].Nume_Producator AS 'Nume
Producator'
FROM [Proiect].[dbo].[Produse] INNER JOIN [Proiect].[dbo].[Producatori]
ON[Proiect].[dbo].[Produse].ID_Producator =
[Proiect].[dbo].[Producatori].ID_Producator
WHERE (lower([Produse].Nume)LIKE '%' + field1.getText() + '%');
```

8. Afisarea tuturor producatorilor introdusi din tastatura

```
SELECT [Nume],[Cantitate],[Pret],[Producatori].Nume_Producator AS 'Nume
Producator'
FROM [Proiect].[dbo].[Produse] INNER JOIN [Proiect].[dbo].[Producatori] ON
[Proiect].[dbo].[Produse].ID_Producator =
[Proiect].[dbo].[Producatori].ID_Producator
WHERE (lower([Producatori].Nume_Producator)LIKE '%' + field1.getText() + '%');
```

9. Selectarea tuturor comenzilor dintr-un oras introdus de la tastatura

```
SELECT [Nume],[CNP],[Sex],[Linia_1] + ' ' + [Linia_2] AS 'Adresa',
[Oras],[Judet],[Cod_Postal]
FROM [Proiect].[dbo].[Client] INNER JOIN [Proiect].[dbo].[Adrese] ON
[Proiect].[dbo].[Client].ID_Adresa = [Proiect].[dbo].[Adrese].ID_Adresa
WHERE lower(' ' + field1.getText() + ' ')IN(SELECT Proiect.dbo.Adrese.Oras
FROM Adrese)
```

10. Afisarea tuturor produselor de pe stoc

```
SELECT [Nume],[Cantitate],[Pret],[Producatori].Nume_Producator AS 'Nume
Producator'
FROM [Proiect].[dbo].[Produse] INNER JOIN [Proiect].[dbo].[Producatori] ON
[Proiect].[dbo].[Produse].ID_Producator =
[Proiect].[dbo].[Producatori].ID_Producator;
```

11. Afisarea produselor care nu au fost cumparate pentru a evita refacerea stocurilor mari

```
SELECT pr.Nume,pr.Pret
FROM Proiect.dbo.Produse pr
WHERE pr.ID_Produs NOTIN(SELECT pc.ID_Produs
FROM Proiect.dbo.Client c1 INNER JOIN Proiect.dbo.Comenzi cm ON c1.ID_Client =
cm.ID_Client INNERJOIN Proiect.dbo.Produse_Comenzi pc on cm.ID_Comanda =
pc.ID_Comanda)
```

12. Afiseaza toate produsele cu pretul mai mare decat pretul mediu al produselor si clientii care le-au cumparat

```
SELECT p.Nume,p.Pret,c1.Nume
FROM Proiect.dbo.Produse p JOIN Proiect.dbo.Produse_Comenzi pc ON p.ID_Produs =
pc.ID_Produs JOIN Proiect.dbo.Comenzi cm ON pc.ID_Comanda = cm.ID_Comanda JOIN
Proiect.dbo.Client c1 ON cm.ID_Client = c1.ID_Client
WHERE p.Pret > (SELECT AVG(p1.Pret) FROM Proiect.dbo.Produse p1)
```

13. Media investitiei, suma totala, numarul de comenzi per client si media sa fie mai mare decat o valoare specificata - pentru fidelizare

```
SELECT c1.Nume, SUM(p2.Pret)/COUNT(p2.ID_Produs) AS 'Medie',SUM(p2.Pret) AS 'Suma
Totala',COUNT(p2.ID_Produs) AS 'Comenzi'
FROM Proiect.dbo.Client c1 JOIN Proiect.dbo.Comenzi cm2 ON c1.ID_Client =
cm2.ID_Client JOIN Proiect.dbo.Produse_Comenzi pc2 ON cm2.ID_Comanda =
pc2.ID_Comanda JOIN Proiect.dbo.Produse p2 ON p2.ID_Produs = pc2.ID_Produs
WHERE 233 <= (SELECT SUM(p.Pret)/COUNT(pc.ID_Produs) FROM Proiect.dbo.Produse p
JOIN Proiect.dbo.Produse_Comenzi pc ON p.ID_Produs = pc.ID_Produs JOIN
Proiect.dbo.Comenzi cm ON pc.ID_Comanda = cm.ID_Comanda WHERE cm.ID_Client =
c1.ID_Client)
GROUP BY c1.Nume
```

Functionalitatea aplicatiei:

Aplicatia are la baza driverul JDBC 6.0 care e inclus in aplicatia scrisa in Java pentru a putea face conexiunea la baza de date. Aplicatia dispune de mai multe scene care sunt redade intr-o fereasta, printre care exista la inceput o scena de Login unde se poate introduce username-ul, parola si baza de date la care se va face conexiunea. Apoi urmeaza o scena unde sunt afisate mai multe butoane pentru a alege pasul urmator, de exemplu, inserarea in baza de date a unor clienti sau stergerea unor produse, respectiv actualizarea produselor.

In scena de actualizare a datelor se poate selecta o tabela din baza de date, apoi se va afisa intr-un tabel respectivele coloane si campuri pentru a facilita actualizarea datelor din baza de date folosindu-se informatiile afisate in tabel.

In scena pentru stergerea datelor se procedeaza similar cu actualizarea datelor.

Scena de inserare a interogarilor in baza de date consta in alegerea unei tabele printr-un buton unde se va face inserarea deschizand o fereasta noua.