Universitatea din București

Facultatea de Matematică și Informatică

Secția Informatică

**Gestiunea unui magazin online**

-proiect Baze de date-

Boghiu Alexandra - Adriana

Grupa 134

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

Proiectul prezintă gestiunea unui magazin online.

Baza de date reține informații cu privire la utilizatori, comenzile, părerile acestora, produse, depozitul unde se află acestea.

Un utilizator locuiește într-o locație (Locație). Acesta poate efectua comenzi (Comanda) care sunt însoțite de facturi (Factură) și livrate printr-o firmă de curierat (Firmă de curierat). Comanda conține produse (Produs) care aparțin unei categorii (Categorie). Există depozite (Depozit) care se află într-o anumită locație (Locație) și care au in stoc produse (Produs). De asemenea, utilizatorul poate lăsa recenzii (Review) despre produse.

**2. Prezentarea constrângerilor (restricții, reguli) impuse asupra modelului.**

Prin entitatea Produs nu mă refer la un produs fizic, la un obiect, ci la mulțimea produselor de acel tip.

Fiecare utilizator care efectuează comenzi are o singură locație, dar pot fi mai mulți utilizatori cu aceeași locație (în cazul unei familii, de exemplu).

Fiecare depozit are o singură locație, dar pot fi mai multe depozite cu aceeași locație.

Utilizatorul nu este obligat să facă vreo comandă. Astfel, în cazul în care un utilizator nu a efectuat nicio comandă, dar și-a creat un cont pe platformă, nu se vor reține date despre locația și numărul său de telefon. Totuși, acesta își poate înscrie datele pe platformă chiar dacă nu efectuează comenzi.

Utilizatorul poate alege pentru fiecare comandă firma de curierat prin care să îi fie livrată comanda, nu este obligat să solicite aceeași firmă de curierat pentru toate comenzile.

Fiecare comandă este însoțită de o singură factură.

Fiecare comandă conține cel puțin un produs.

Un produs aparține unei singure categorii.

Pot exista mai multe review-uri (sau niciunul) pentru un produs.

Un utilizator poate scrie mai multe review-uri.

Un produs se poate regăsi în mai multe depozite, iar fiecare depozit poate avea produse din categorii diferite în stoc.

**3. Descrierea entităților, incluzând precizarea cheii primare.**

Entitățile pe care am ales să le includ sunt: Utilizator, Comandă, Factură, Firmă de curierat, Locație, Depozit, Produs, Categorie, Review.

Utilizator:

* Cheie primară: id\_utilizator
* Include date despre acesta precum Nume, Prenume, Email, Telefon
* Persoană fizică
* Locuiește într-o locație
* Poate să efectueze comenzi
* Poate să scrie review-uri

Comandă:

* Reține date cu privire la comandă
* Cheie primară: id\_comanda
* Include date despre data comenzii, status
* Este însoțită de o factură
* Conține produse

Factură:

* Reține date cu privire la factură
* Cheie primară: id\_factura
* Este unică pentru fiecare comandă

Firmă de curierat:

* Reține informații cu privire la firma de curierat care se ocupă de o anumită comandă
* Cheie primară: id\_curier

Locație:

* Reține date cu privire la locație
* Cheie primară: id\_locatie

Depozit:

* Reține date cu privire la depozit
* Cheie primară: id\_depozit
* Se află la o locație
* Are produse în stoc

Produs:

* Reține date cu privire la produs
* Cheie primară: id\_produs
* Aparține unei categorii

Categorie:

* Reține denumirea categoriei din care face parte un anumit produs
* Cheie primară: id\_categorie

Review:

* Reține review-ul lăsat de un utilizator pentru un anumit produs (număr de stele, comentariu, dată)
* Cheie primară: id\_review
* Chei străine: id\_utilizator, id\_produs

**4.Descrierea relațiilor, incluzând precizarea cardinalității acestora.**

UTILIZATOR\_*locuieste*\_LOCATIE = relație de tip *one-to-many* între entitățile Utilizator și Locație ce reflectă legătura dintre acestea (ce utilizatori locuiesc într-o anumită locație). Cardinalitatea minima este *0:1*, iar cardinalitatea maximă *1:m*. Un utilizator are o singură locație, dar mai mulți utilizatori pot locui într-o locație.

UTILIZATOR\_*efectuează*\_COMANDA = relație de tip *one-to-many* între entitățile Utilizator și Comandă ce reflectă legătura dintre acestea (ce comenzi efectuează un anumit utilizator). Cardinalitatea minima este *1:0*, iar cardinalitatea maximă *1:m*. O comandă poate fi efectuată de un singur utilizator, dar un utilizator poate sa efectueze mai multe comenzi (sau niciuna).

UTILIZATOR\_*solicita*\_FIRMA\_DE\_CURIERAT = relație de tip *many-to-many* între entitățile Utilizator și Firmă de curierat ce reflect legătura dintre acestea (ce firme de curierat solicit utilizatorul). Cardinalitatea minima este *0:0*. Cardinalitatea maximă este *m:n*. Un utilizator nu este obligat să solicite o firmă de curierat (în caz că nu realizează comenzi), iar o firmă de curierat poate să nu fie solicitată. Utilizatorul poate să solicite firme diferite de curierat pentru comenzile sale.

COMANDA\_*este\_insotita\_de*\_FACTURA = relație de tip *one-to-one* între entitățile Comandă și Factură. Fiecare comandă este însoțită de o singură factură, iar fiecare factură însoțeste o singură comandă.

COMANDA\_*este\_livrata\_de*\_FIRMA\_DE\_CURIERAT = relație de tip *one-to-many* între entitățile Comandă și Firmă de curierat. Cardinalitatea minima este *0:1*, iar cea maximă *1:m*. O firmă de curierat poate să livreze mai multe comenzi, iar fiecare comandă are o singură firmă de curierat asociată.

COMANDA\_*contine\_*PRODUS = relație de tip *many-to-many* între entitățile Comandă și Produs, reflectând legătura dintre acestea (ce produse conține o comandă). Cardinalitatea minimă este *0:1*, iar cea maximă *m:n.* O comandă poate conține mai multe produse, iar un produs poate fi inclus în mai multe comenzi.

PRODUS\_*apartine*\_CATEGORIE = relație de tip *many-to-one* între entitățile Produs și Categorie, reflectând legătura dintre acestea (ce produse aparțin unei categorii). Fiecare produs aparține unei singure categorii, iar fiecare categorie poate include mai multe produse.

UTILIZATOR\_*scrie*\_REVIEW = relație de tip *many-to-one* între entitățile Utilizator și Review, reflectând legătura dintre acestea (ce review-uri scrie un utilizator). Cardinalitatea minima este 1:0, iar cea maximă 1:m. Un utilizator poate scrie mai multe review-uri, iar fiecare review este asociat unui utilizator.

REVIEW\_*corespunde*\_PRODUS = relație de tip *many-to-one* între entitățile Review și Produs. Cardinalitatea minima este 1:0, iar cea maximă 1:m. Un produs poate să aibă mai multe review-uri, iar fiecare review corespunde unui produs.

DEPOZIT\_*se\_afla\_la*\_LOCATIE = relație de tip *many-to-one* între entitățile Depozit și Locatie, reflectând legătura dintre acestea (ce locație îi este asociată fiecărui depozit). Cardinalitatea minima este 1:0, iar cea maximă 1:m.

DEPOZIT\_*are\_in\_stoc*\_PRODUS = relație de tip *many-to-many* între entitățile Depozit și Produs, reflectând legătura dintre acestea (ce produse are în stoc un depozit). Cardinalitatea minima este 0:0, iar cea maximă m:m. Un produs poate să nu fie pe stoc în niciun depozit.

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

Entitatea Utilizator are atributele:

* id\_utilizator: cheie primară, variabilă nenulă de tip numeric, de lungime maximă 4, reprezintă codul unui utilizator
* Nume: variabilă nenulă de tip caracter, de lungime maximă 20, reprezintă numele utilizatorului
* Prenume: variabilă nenulă de tip caracter, de lungime maximă 20, reprezintă prenumele utilizatorului
* Email: : variabilă nenulă de tip caracter, de lungime maximă 30, reprezintă adresa de email a utilizatorului
* Telefon: variabilă de tip numeric, de lungime maximă 10, reprezintă numărul de telefon al utilizatorului
* id\_locatie: variabilă de tip numeric, de lungime maximă 3, reprezintă locația utilizatorului; atributul corespunde unei valori a cheii primare din tabelul Locație

Entitatea Comandă are atributele:

* id\_comanda: cheie primară, variabilă nenulă de tip numeric, de lungime maximă 5, reprezintă codul unei comenzi
* Data: variabilă de tip dată, reprezintă data comenzii
* Status: variabilă de tip caracter, de lungime maximă 10, care poate lua valorile: Livrat, Expediat, Procesat, reprezintă statusul comenzii

Entitatea Factură are atributele:

* id\_factură: cheie primară, variabilă nenulă de tip numeric, de lungime maximă 5, reprezintă codul unei facturi
* Data\_facturare: variabilă de tip dată, reprezintă data în care s-a efectuat facturarea
* id\_comanda: variabilă de tip numeric, de lungime maximă 5, leagă factura de comanda căreia îi corespunde

Entitatea Firmă\_de\_curierat are atributele:

* id\_curier: cheie primară, variabilă nenulă de tip numeric, de lungime maximă 1, reprezintă codul firmei de curierat
* Denumire: variabilă nenulă de tip caracter, de lungime maximă 10 ce poate lua valorile: Cargus, FAN, GLS, Sameday, DHL; reprezintă denumirea firmei de curierat
* Telefon: variabilă nenulă de tip numeric, de lungime maximă 10, reprezintă numărul de telefon al utilizatorului

Entitatea Locație are atributele:

* id\_locatie: cheie primară, variabilă nenulă de tip numeric, de lungime maximă 3, reprezintă codul locației
* Judet: variabilă nenulă de tip caracter, de lungime maximă 10, reprezintă numele județului
* Oras: variabilă nenulă de tip caracter, de lungime maximă 10, reprezintă numele orașului
* Strada: variabilă nenulă de tip caracter, de lungime maximă 20, reprezintă numele străzii
* Numar: variabilă de tip numeric, de lungime maximă 3, reprezintă numărul străzii

Entitatea Depozit are atributele:

* id\_depozit: cheie primară, variabilă nenulă de tip numeric, de lungime maximă 4, reprezintă codul depozitului
* Email: variabilă nenulă de tip caracter, de lungime maximă 30, reprezintă adresa de email a depozitului
* Telefon: variabilă nenulă de tip numeric, de lungime maximă 10, reprezintă numărul de telefon al utilizatorului
* id\_locație: variabilă de tip numeric, de lungime maximă 3; atributul corespunde unei valori a cheii primare din tabelul Locație

Entitatea Produs are atributele:

* id\_produs: cheie primară, variabilă nenulă de tip numeric, de lungime maximă 5, reprezintă codul produsului
* Denumire\_produs: variabilă nenulă de tip caracter, de lungime maximă 20, reprezintă numele produsului
* Descriere: variabilă de tip caracter, de lungime maximă 50, reprezintă o scurtă descriere a produsului (de exemplu, culoare, material, etc)
* Preț: variabilă nenulă de tip numeric, de lungime maximă 5, reprezintă prețul unui astfel de produs
* Rating: variabilă de tip numeric, cu lungimea maximă 1, care poate lua valorile 0, 1, 2, 3, 4 sau 5; media stelelor pe care le-a primit produsul
* id\_categorie: variabilă nenulă de tip numeric, de lungime maximă 5; atributul corespunde unei valori a cheii primare din tabelul Categorie

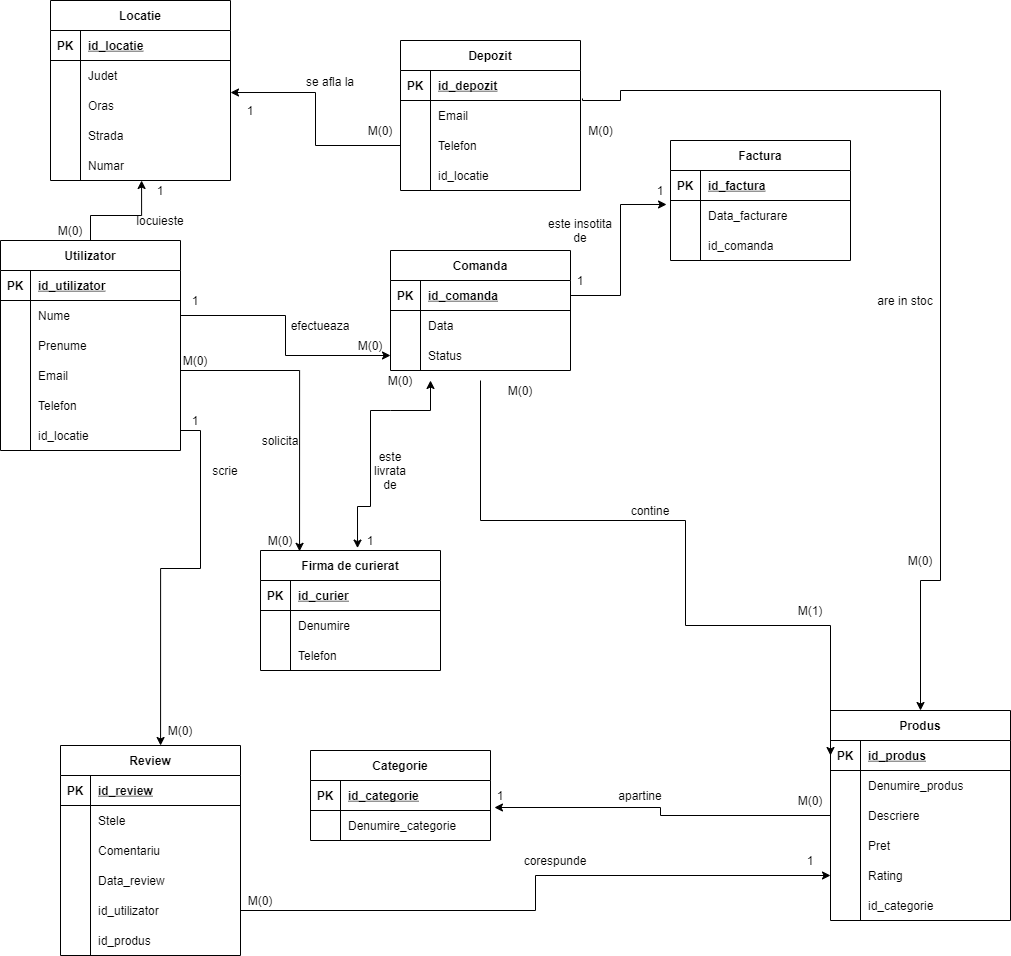
Entitatea Categorie are atributele:

* id\_categorie: cheie primară, variabilă nenulă de tip numeric, de lungime maximă 5
* Denumire\_categorie: variabilă nenulă de tip caracter, de lungime maximă 20, reprezintă numele categoriei

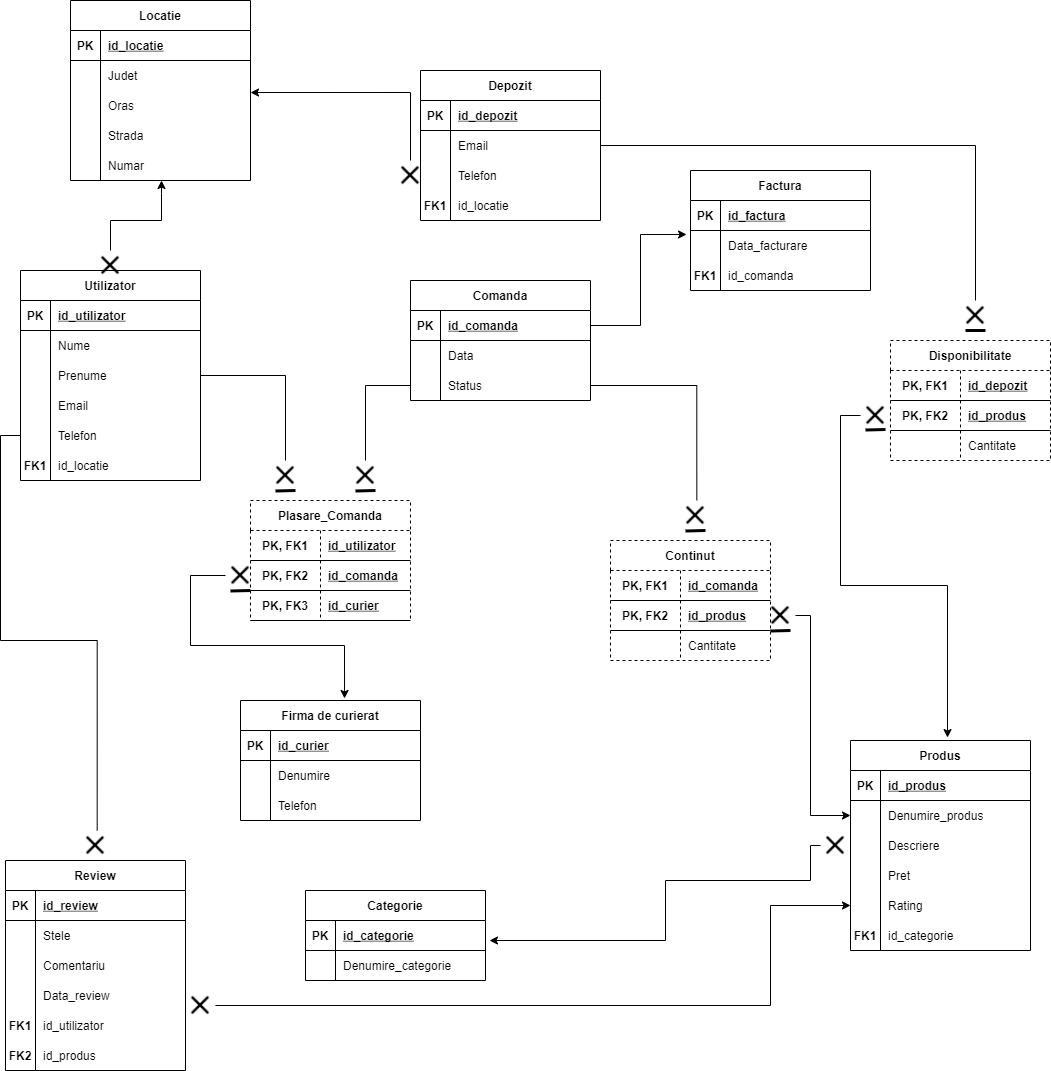
Entitatea Review are atributele:

* id\_review: cheie primară, variabilă nenulă de tip numeric, de lungime maximă 3, reprezintă codul review-ului
* Stele: variabilă nenulă de tip numeric, cu lungimea maximă 1, care poate lua valorile 0, 1, 2, 3, 4 sau 5; reprezintă numărul de stele pe care l-a oferit utilizatorul produsului
* Comentariu: variabilă de tip caracter, de lungime maximă 100, scurt comentariu lăsat de utilizator
* Data\_review: variabilă nenulă de tip dată, reprezintă data review-ului
* id\_utilizator: variabilă de tip numeric, de lungime maximă 4, leagă utilizatorul de review-ul pe care îl lasă
* id\_produs: variabilă de tip numeric, de lungime maximă 5, reprezintă codul produsului, leagă review-ul de produsul corespunzător

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

****

**7.Realizarea diagramei conceptuale corespunzătoare diagramei entitate-relație proiectate la punctul 6. Diagrama conceptuală obținută trebuie să conțină minimum 6 tabele (fără considerarea subentităților), dintre care cel puțin un tabel asociativ.**

****

**8. Enumerarea schemelor relaționale corespunzătoare diagramei conceptuale proiectate la punctul 7.**

Utilizator (id\_utilizator#, nume, prenume, email, telefon, id\_locatie)

Locatie (id\_locatie#, judet, oras, strada, numar)

Depozit (id\_depozit#, email, telefon, id\_locatie)

Disponibilitate (id\_depozit#, id\_produs#, cantitate)

Produs (id\_produs#, denumire\_produs, descriere, pret, rating, id\_categorie)

Categorie (id\_categorie#, denumire\_categorie)

Review (id\_review#, stele, comentariu, data\_review, id\_utilizator, id\_produs)

Continut (id\_comanda#, id\_produs#, cantitate)

Plasare\_Comanda (id\_utilizator#, id\_comanda#, id\_curier#)

Firma de curierat (id\_curier#, denumire, telefon)

Comanda (id\_comanda#, data, status)

Factura (id\_factura#, data\_facturare, id\_comanda)

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

Menționez că valorile introduse în tabele nu sunt aceleași cu cele pe care urmează să le introduc la punctul 10, ci au numai scopul de a exemplifica normalizarea.

Non-FN1:

Utilizator (id\_utilizator#, nume, prenume, email, telefon, id\_locatie)

Locatie (id\_locatie#, judet, oras, strada, numar)

Depozit (id\_depozit#, email, telefon, id\_locatie)

Disponibilitate (id\_depozit#, id\_produs#, cantitate)

Produs (id\_produs#, denumire\_produs, descriere, pret, rating, id\_categorie)

Categorie (id\_categorie#, denumire\_categorie)

Review (id\_review#, continut, data\_review, id\_utilizator, id\_produs)

Continut (id\_comanda#, id\_produs#, cantitate, status)

Plasare\_Comanda (id\_utilizator#, id\_comanda#, id\_curier#)

Comanda (id\_comanda#, data, denumire\_curier, telefon\_curier)

Factura (id\_factura#, data\_facturare, id\_comanda)

Review

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id\_review# | continut | data\_review | id\_utilizator | id\_produs |
| 1 | 2 stele produs slab | 13.06.2018 | 3 | 5 |
| 2 | 1 stea nu recomand | 26.10.2019 | 5 | 5 |
| 3 | 5 stele perfect | 19.07.2019 | 3 | 2 |
| 4 | 4 stele foarte bun | 20.08.2020 | 6 | 4 |
| 5 | 5 stele recomand | 21.03.2021 | 2 | 1 |

Nu se află în FN1, deoarece atributul ,,continut” nu este atomic. Acesta include atât numărul de stele, cât și comentariul. Pentru a aduce la FN1, este de ajuns să înlocuim atributul ,,continut” cu atributele ,,stele” și ,,comentariu” în următorul mod:

Review

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| id\_review# | stele | comentariu | data\_review | id\_utilizator(FK1) | id\_produs(FK2) |
| 1 | 2 | produs slab | 13.06.2018 | 3 | 5 |
| 2 | 1 | nu recomand | 26.10.2019 | 5 | 5 |
| 3 | 5 | perfect | 19.07.2019 | 3 | 2 |
| 4 | 4 | foarte bun | 20.08.2020 | 6 | 4 |
| 5 | 5 | recomand | 21.03.2021 | 2 | 1 |

FN1 și Non-FN2:

Utilizator (id\_utilizator#, nume, prenume, email, telefon, id\_locatie)

Locatie (id\_locatie#, judet, oras, strada, numar)

Depozit (id\_depozit#, email, telefon, id\_locatie)

Disponibilitate (id\_depozit#, id\_produs#, cantitate)

Produs (id\_produs#, denumire\_produs, descriere, pret, rating, id\_categorie)

Categorie (id\_categorie#, denumire\_categorie)

Review (id\_review#, stele, comentariu, data\_review, id\_utilizator, id\_produs)

Continut (id\_comanda#, id\_produs#, cantitate, status)

Plasare\_Comanda (id\_utilizator#, id\_comanda#, id\_curier#)

Comanda (id\_comanda#, data, denumire\_curier, telefon\_curier)

Factura (id\_factura#, data\_facturare, id\_comanda)

Atributul ,,status” depinde numai de ,,id\_comanda”, nu și de ,,id\_produs”. Pentru a aduce la FN2, putem muta atributul în ,,Comanda”.

FN2 și Non-FN3:

Utilizator (id\_utilizator#, nume, prenume, email, telefon, id\_locatie)

Locatie (id\_locatie#, judet, oras, strada, numar)

Depozit (id\_depozit#, email, telefon, id\_locatie)

Disponibilitate (id\_depozit#, id\_produs#, cantitate)

Produs (id\_produs#, denumire\_produs, descriere, pret, rating, id\_categorie)

Categorie (id\_categorie#, denumire\_categorie)

Review (id\_review#, stele, comentariu, data\_review, id\_utilizator, id\_produs)

Continut (id\_comanda#, id\_produs#, cantitate)

Plasare\_Comanda (id\_utilizator#, id\_comanda#, id\_curier#)

Comanda (id\_comanda#, data, status, denumire\_curier, telefon\_curier)

Factura (id\_factura#, data\_facturare, id\_comanda)

Comanda

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id\_comanda# | data | status | denumire\_curier | telefon\_curier |
| 1 | 23.01.2018 | Livrat | Urgent Cargus | 0770123456 |
| 4 | 04.02.2018 | Livrat | Urgent Cargus | 0770123456 |
| 6 | 25.04.2019 | Livrat | Sameday | 0770987654 |
| 7 | 29.05.2021 | În curs de livrare | GLS | 0770234567 |

Atributul ,,telefon\_curier” depinde de ,,denumire\_curier”. Astfel, pentru a aduce la FN3 putem adăuga un nou tabel cu informații despre firma de curierat care se ocupă de livrarea comenzii.

Comanda

|  |  |  |
| --- | --- | --- |
| id\_comanda# | data | status |
| 1 | 23.01.2018 | Livrat |
| 4 | 04.02.2018 | Livrat |
| 6 | 25.04.2019 | Livrat |
| 7 | 29.11.2020 | În curs de livrare |

Firma\_de\_curierat

|  |  |  |
| --- | --- | --- |
| id\_curier# | denumire\_curier | telefon\_curier |
| 1 | Urgent Cargus | 0770123456 |
| 4 | Urgent Cargus | 0770123456 |
| 6 | Sameday | 0770987654 |
| 7 | GLS | 0770234567 |

FN3:

Utilizator (id\_utilizator#, nume, prenume, email, telefon, id\_locatie)

Locatie (id\_locatie#, judet, oras, strada, numar)

Depozit (id\_depozit#, email, telefon, id\_locatie)

Disponibilitate (id\_depozit#, id\_produs#, cantitate)

Produs (id\_produs#, denumire\_produs, descriere, pret, rating, id\_categorie)

Categorie (id\_categorie#, denumire\_categorie)

Review (id\_review#, stele, comentariu, data\_review, id\_utilizator, id\_produs)

Continut (id\_comanda#, id\_produs#, cantitate)

Plasare\_Comanda (id\_utilizator#, id\_comanda#, id\_curier#)

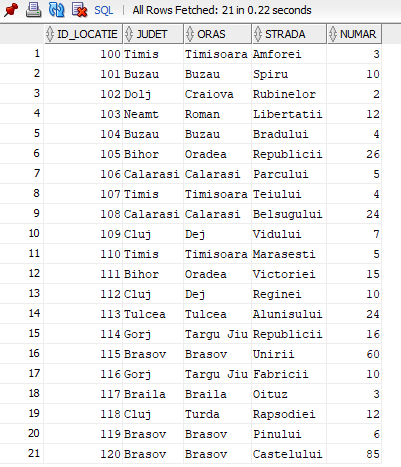
Firma de curierat (id\_curier#, denumire, telefon)

Comanda (id\_comanda#, data, status)

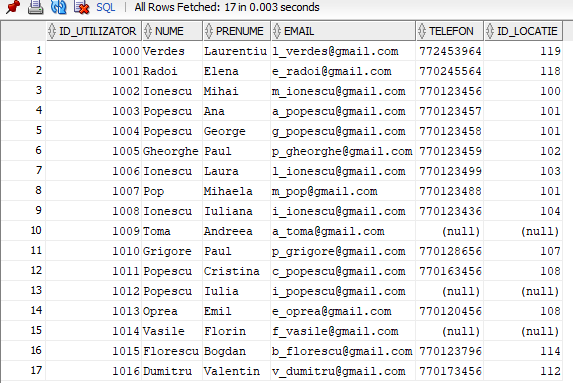
Factura (id\_factura#, data\_facturare, id\_comanda)

**10. Crearea tabelelor în SQL și inserarea de date coerente în fiecare dintre acestea (minimum 5 înregistrări în fiecare table neasociativ; minimum 10 înregistrări în tabelele asociative).**

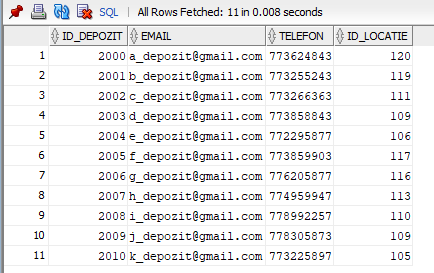
Locație:

****

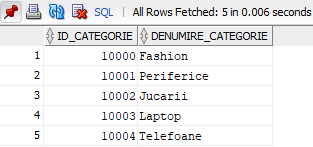
Utilizator:



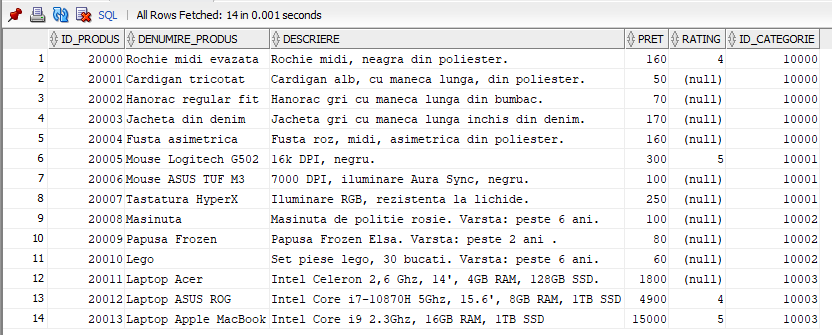
Depozit:

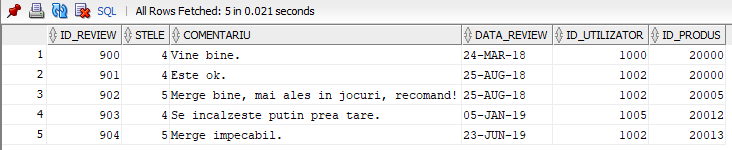


Categorie:

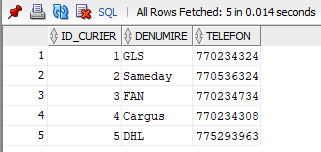


Produs:

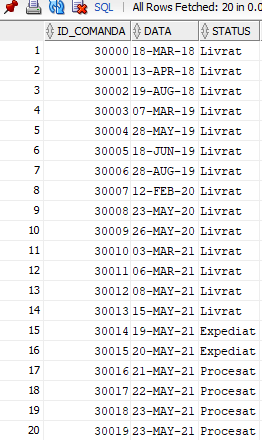


Review: 

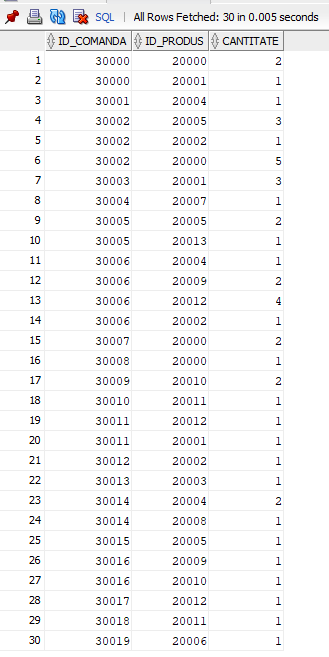
Firma\_de\_curierat:



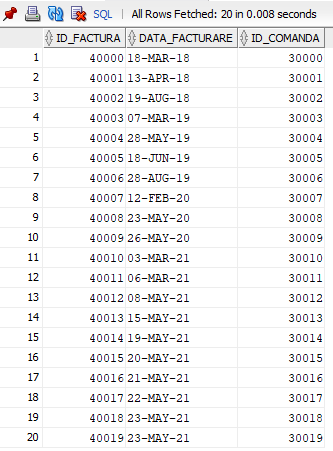
Comanda:



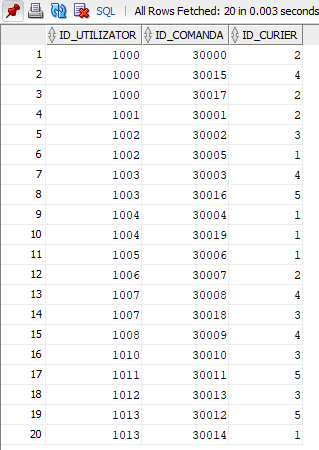
Continut:



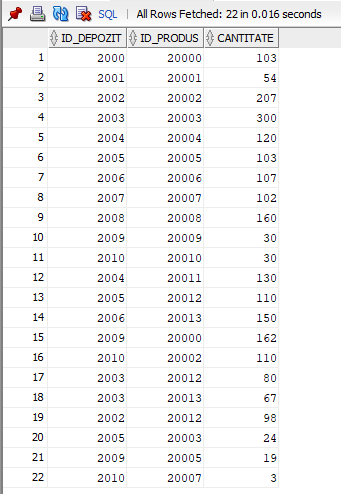
Factura:



Plasare\_Comanda:



Disponibilitate:



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

**•operație join pe cel puțin 4 tabele**

**•filtrare la nivel de linii**

**•subcereri sincronizate în care intervin cel puțin 3 tabele**

**•subcereri nesincronizate în care intervin cel puțin 3 tabele**

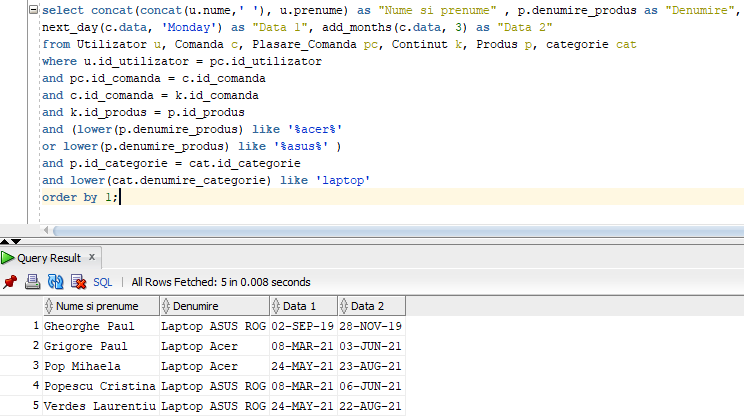
**•grupări de date, funcții grup, filtrare la nivel degrupuri**

**•ordonări**

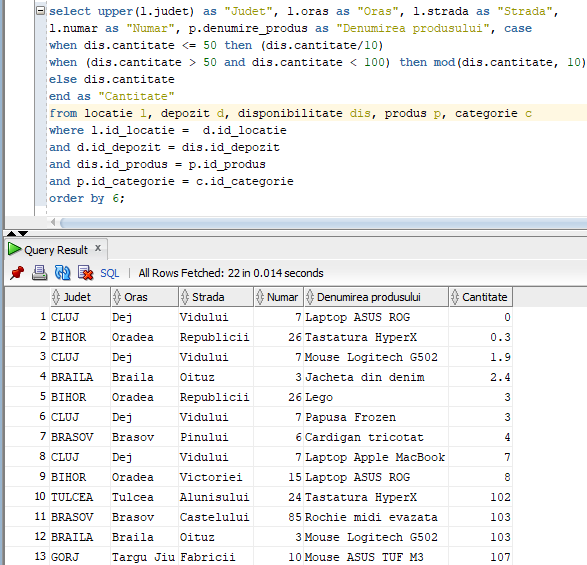
**•utilizarea a cel puțin 2 funcții pe șiruri de caractere, 2 funcții pe date calendaristice, a funcțiilor NVL și DECODE, a cel puțin unei expresii CASE**

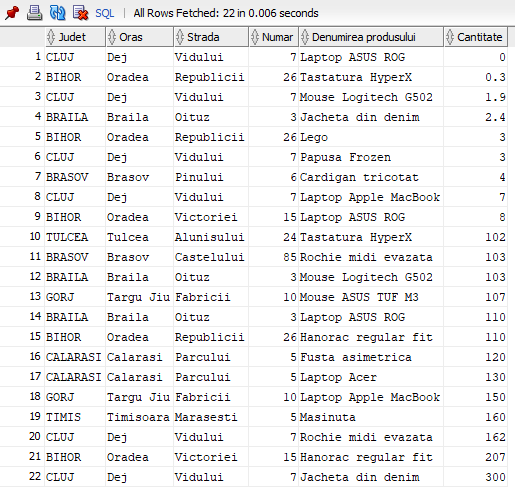
**•utilizarea a cel puțin 1 bloc de cerere(clauza WITH)**

1. Să se afișeze în ordine alfabetică numele și prenumele utilizatorilor (într-o singura coloana) care au comandat un laptop Asus sau Acer, cât și modelul de laptop (numele produsului), următoarea zi de luni după data comenzii și data comenzii, adăugându-i 3 luni.

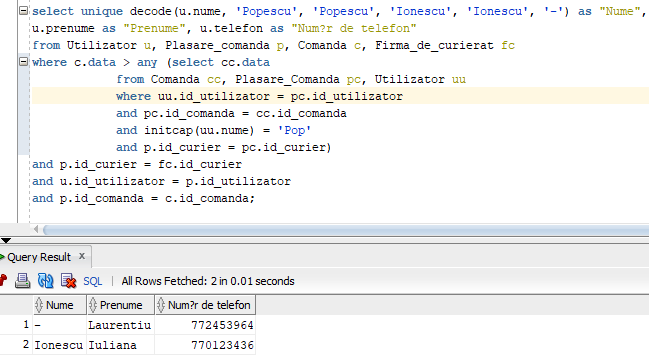


1. Să se afișeze adresa completă a depozitelor și denumirea produselor din acestea. Afișati județul cu majuscule. În cazul în care cantitatea unui produs dintr-un depozit este mai mică sau egală cu 50, se va afișa prima cifra a numărului, dacă este cuprinsă între 50 și 100 se va afișa ultima cifră, altfel se va afișa cantitatea fără modificări. Ordonați crescător după valorile cantităților.

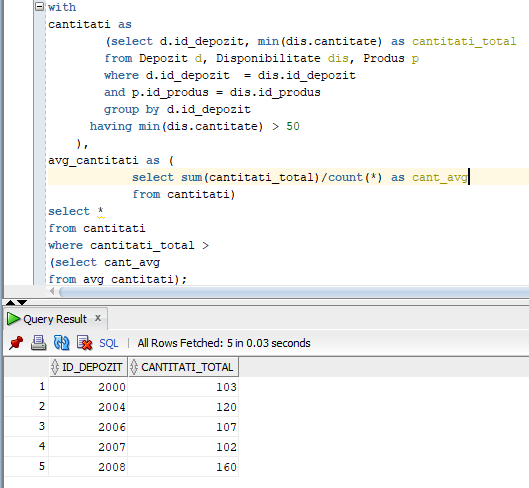




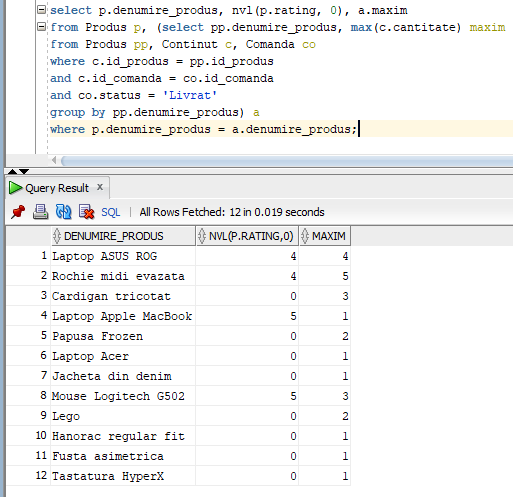
1. Să se afișeze numele, prenumele și numărul de telefon al utilizatorilor care au făcut comenzi după prima comandă efectuată de utilizatorul cu numele Pop (inclusiv acesta, dacă a efectuat alte comenzi ulterior) prin același curier, fără duplicate. Dacă utilizatorul curent nu se numește Popescu sau Ionescu, se va afișa o linie ('-') în locul numelui.



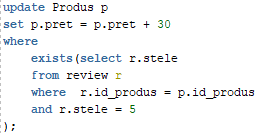
4) Să se afișeze codul departamentelor și cantitatea totală. Se vor considera departamentele ale căror cantități nu sunt mai mici decât 50, iar cantitatea minimă este peste medie.



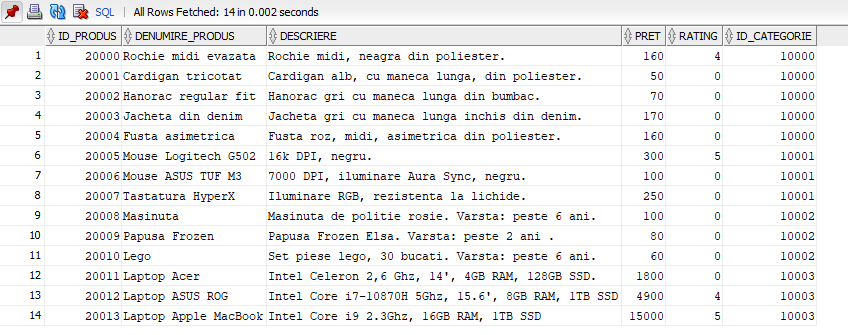
5) Să se afișeze denumirea produselor, rating-ul, numărul maxim de bucăți dintre toate comenzile deja livrate care includ produsul. Daca produsul nu are rating (este null), se va afișa 0.

**12.Implementarea a 3 operații de actualizare sau suprimare a datelor utilizând subcereri.**

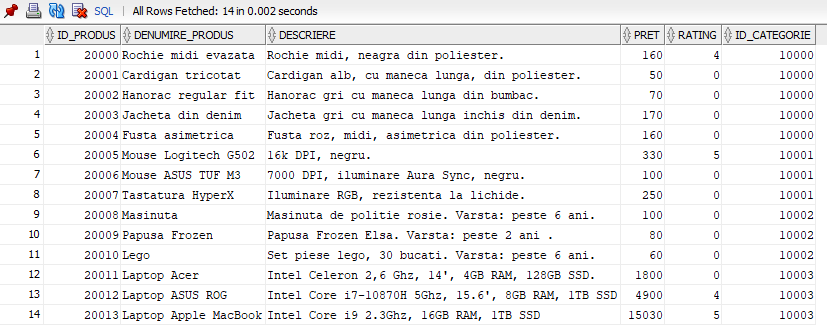
1. Să se mărească cu 30 de lei toate produsele care au cel puțin un review de 5 stele.



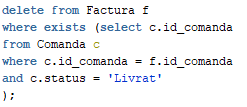
Tabelul Produs înainte:



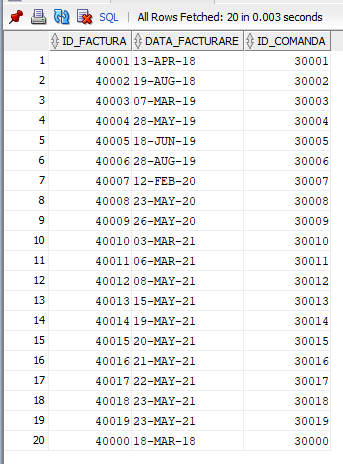
Tabelul Produs după:



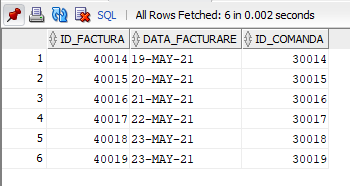
1. Să se șteargă toate facturile pentru comenzi deja livrate.



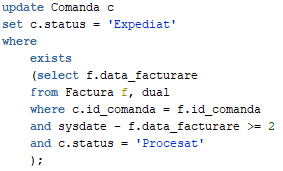
Tabelul Factura înainte:



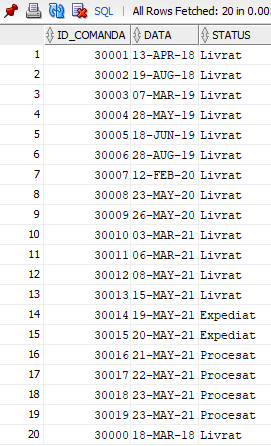
Tabelul Factură după:



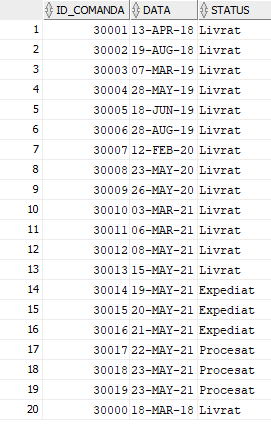
1. Să se schimbe statusul comenzilor cu status Procesat în Expediat dacă data lor de facturare este mai veche de 2 zile.



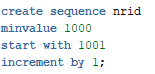
Tabelul Comanda înainte:



Tabelul Comanda după:



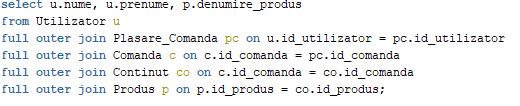
**13. Crearea unei secvențe ce va fi utilizată în inserarea înregistrărilor în tabele (punctul 10).**



Secvența a fost folosită în inserarea înregistrărilor în tabelul Utilizator.

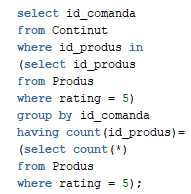
**16. Formulați în limbaj natural și implementați în SQL: o cerere ce utilizează operația *outer-join* pe minimum 4 tabele și două cereri ce utilizează operația *division*.**

1. Să se afișeze numele, prenumele tuturor utilizatorilor, cât și produsele comandate. Se vor afișa și cei care nu au efectuat comenzi.





2) Să se afișeze codurile comenzilor care conțin toate produsele cu rating 5.





3) Să se afișeze codurile comenzilor care conțin toate produsele cu prețul 60.

