

UNIVERSITATEA DIN BUCUREŞTI
FACULTATEA DE MATEMATICĂ ŞI INFORMATICĂ
SPECIALIZAREA INFORMATICĂ

Sistem de gestiune a datelor unei firme de retail de mobilier

Proiect SQL realizat de Alexandrescu Andra

Seria 23, Grupa 232, Anul Universitar curent 2024-2025

PROF. LABORATOR: EDUARD SZMETEANCA
PROF. CURS: GABRIELA MIHAI

Contents

Specificații mediu lucru	4
Descriere Model.....	4
Utilitate model	4
Reguli de funcționare	4
Constrângeri asupra modelului	5
Entități.....	6
Relații.....	8
Atribute	11
Diagrama entitate-relație	16
Diagrama conceptuală	17
Schemele relaționale corespunzătoare diagramei conceptuale	18
Realizarea normalizării până la FN ₃	18
Proiectare stadiu de început abandonată	20
Reconfigure project	20
Secvențe pentru inserarea înregistrărilor în tabele	21
Crearea tabelelor și inserarea datelor.....	23
Modificare datetime și timestamp format	23
Creare tabele.....	24
Inserare date	30
Tabela CLIENT	41
Tabela CLIENT_ONLINE	41
Tabela CLIENT_FIZIC	41
Tabela ADRESA.....	42
Tabela MAGAZIN	42
Tabela AGENT_VANZARI	42
Tabela COMANDA	43
Tabela TRANZACTIE.....	44
Tabela CATEGORIE.....	44
Tabela PIESA_MOBILIER	45
Tabela ADAUGA_COMANDA	46
Tabela OFERTA.....	47
Tabela FURNIZOR.....	47
Tabela MATERIE_PRIMA.....	48
Tabela PRODUSA_DIN.....	48
Tabela STOC	49

Tabela APROVIZIONEAZA.....	49
Adăugare tabela DETALII_PRODUSE pentru exercițiul 10 (pe baza datelor din tabelele din DB)	49
Exercițiul 6	50
Cerință	50
Implementare	50
DBMS OUTPUT.....	54
Exercițiul 7	56
Cerință	56
Implementare	56
DBMS OUTPUT.....	60
Exercițiul 8	61
Cerință	61
Implementare	61
DBMS OUTPUT.....	65
Exercițiul 9	66
Cerință	66
Implementare	66
DBMS OUTPUT.....	71
Idee intersecție prin interclasare abandonată exercițiul 9 (din cauză că ar fi o filtrare decât să trateze cazuri definite pe baza no_data_found)	74
Cerință	74
Implementare	74
DBMS OUTPUT.....	79
Exercițiul 10	81
Cerință	81
Implementare	81
OUTPUT	83
Exercițiul 11.....	85
Cerință	85
Implementare	85
OUTPUT	89
Exercițiul 12	92
Cerință	92
Implementare	92
OUTPUT	94
DBMS OUTPUT.....	95

Exercițiul 13.....	96
Cerință	96
Implementare	96
DBMS OUTPUT.....	102

Specificații mediu lucru

Oracle Database 19c

Microsoft Windows x64

Windows specifications: Windows 10 Enterprise

fără VM

Descriere Model

Utilitate model

Proiectul are ca scop descrierea proceselor din cadrul unei companii de retail care comercializează mobilier, modelul de gestiune fiind o reprezentare a unei modalități de funcționare a unei afaceri de acest tip. Baza de date concepută se bazează pe cea a mobilierului de la IKEA și are rolul să analizeze fluxul de lucru, ce presupune procese complexe, precum producția produselor de mobilier, gestionarea logisticii, procesarea comenzi și a tranzacțiilor.

Reguli de funcționare

Modelul real implică o firma de mobilier care vînde piese de mobilier în mai multe magazine aflate în locații/ adrese diferite, fiind showroom-uri sau centre de ridicare comenzi, însemnând atât că vor vînde produse, cât și că vor reprezenta posibile adrese de livrare a comenziilor clientilor.

Firma vînde piese de mobilier produse în propriile fabrici sau produse de persoane terțe, existând informații despre materialele folosite în procesul de fabricație al unui produs numai dacă acesta este fabricat de către companie. Firma colaborează cu furnizori pentru obținerea materiei prime necesare în fabricarea unei piese de mobilier și cu toate că un model de afaceri real are angajați de mai multe profesii, reprezentativ pentru desfășurarea activității comerciale menționate este numai agentul de vânzări, care procesează comenziile clientilor, lucrând online sau fizic la o singură sucursală a companiei, însemnând că îl poate fi asociat cel mult un magazin.

Fiecare piesă de mobilier aparține unei categorii, cu scopul de a identifica produsele în funcție de anumite caracteristici, iar pentru realizarea produsului final sunt folosite materiale sau materii prime aprovizionate de furnizori. Pot fi asociate oferte fiecărei piese de mobilier într-o perioadă de timp stabilită.

Pentru a asigura că fiecare produs este valabil, evidența mobilierului în cadrul fiecărui magazin este păstrată la nivelul stocului, acest fapt însemnând că un stoc poate fi accesat de mai multe magazine. Relația care se stabileste între aceste trei entități are rolul de a răspunde la întrebarea „Câte piese de mobilier are un magazin distribuite la nivelul stocurilor asociate?”. Se dorește acest lucru pentru a ține evidența distribuirii comune a mărfurii la nivelul mai multor locații ale firmei, cu scopul de a putea aprovizia regulat și de a analiza raportul dintre produse disponibile și vândute într-o perioadă de

timp. Stocurile vor conține mai multe piese de mobilier și se vor identifica, în afară de codul stocului, prin data aprovizionării.

Un client poate plasa comenzi la care să adauge piese de mobilier. De asemenea, clientul are posibilitatea să solicite livrarea unei comenzi la o adresă specificată. O comandă e finalizată printr-o tranzacție care are informații despre plata efectuată.

În plus, pentru realizarea comenzi, clientul e identificat după modul în care își achiziționează produsele (fizic sau online) și după tipul juridic (persoană fizică sau juridică), deoarece comanda/factura reflectă normele de practică ale firmelor de mobilă și bricolaje. Dacă clientul efectuează o comandă online va fi nevoie să completeze niște informații suplimentare specifice unui utilizator al platformei firmei. Unui client fizic îi este solicitat numai numărul de telefon, în cazul în care optează pentru serviciile de livrare. Referitor la tipul legal al unui client, nu este o proprietate imediat specificată, această clasificare este conceptuală, în sensul că un client de tip juridic va avea completat câmpul nume_firma, în timp ce unul fizic nu îl va avea.

Constrângeri asupra modelului

- O comandă poate să fie realizată fizic sau online, motiv pentru care există două categorii de clienți: client_fizic și client_online.
- Un agent de vânzări procesează mai multe comenzi sau niciuna. Fiecare comandă poate să fie validată de către un singur agent sau de niciunul.
- Un agent de vânzări lucrează la cel mult un magazin sau niciunul.
- O comandă poate fi livrată la cel mult o adresă sau niciuna. Comanda trebuie livrată la o adresă în cazul în care clientul optează pentru livrarea acesteia.
- O tranzacție corespunde exact unei comenzi și vice versa. Modalitatea de plată este de două tipuri: „cash” sau „card”. Statusul plății este de patru tipuri: „în procesare”, „verificare”, „aprobată”, „respinsă”, indiferent de tipul plății.
- Un magazin se află exact la o adresă/ locație. Atributul de centru_ridicare al magazinului ia una dintre valorile 0 sau 1, 0 însemnând că e un showroom, iar 1 că e un centru de ridicare. Un magazin are mai mulți agenți de vânzări sau niciunul.
- O adresă este asociată mai multor comenzi sau niciuneia. O adresă aparține cel mult unui magazin sau niciunui.
- Un stoc e asociat mai multor magazine sau niciunuia. Un stoc conține mai multe piese de mobilier sau niciuna. Un magazin are asociat mai multe stocuri sau niciunul. Un magazin vinde mai multe piese de mobilier sau niciuna. O piesă de mobilier se află în mai multe stocuri sau niciunul. O piesă de mobilier e vândută la mai multe magazine sau niciunul.
- O materie primă este aprovizionată de către exact un furnizor. Nu orice piesă de mobilier este produsă de către firma, nu întotdeauna fiind cunoscute materialele, alături de cantitățile folosite și prețul per unitate, deoarece unele produse provin de la terți. Această alegere nu respectă modelul real al magazinului IKEA.
- Nu există garanția că o piesă de mobilier va avea asociat cel puțin o materie primă, adică o piesă de mobilier este produsă din mai multe materii prime sau niciuna.
- O piesă de mobilier are mai multe oferte sau niciuna. Nu există oferte la același produs care să înceapă de pe aceeași data succesiv (unicitatea cheii).
- Nu există înregistrări pentru stocuri care să fie aprovizionate în aceeași data calendaristică (unicitatea cheii).
- La o comandă la care a fost adăugat un produs, nu există posibilitatea de a-l adăuga încă o data la comandă.

Entități

1) AGENT_VANZARI

Un agent de vânzări în cazul modelului real modelat sugerează un angajat care procesează și verifică comenziile clienților în regim fizic (casier), adică îi este asociat un magazin, sau online (consultant, reprezentant vânzări), nu îi este asociat un magazin.

Cheia primară este reprezentată prin id_angajat.

Este o entitate dependentă de MAGAZIN, care are participare optională.

Reguli relații:

- *Nu toate comenziile sunt procesate de către un agent de vânzări, nu există obligativitatea procesării, atât comenziile realizate online, cât și cele fizice, într-un magazin, pot avea asociate un angajat sau niciunul.*
- *Nu toți agenții de vânzări lucrează în regim fizic la un magazin, pot să lucreze și în regim online, referitor la asocierea unui magazin.*

2) CLIENT

Un individ care achiziționează piese de mobilier prin efectuarea unei comenzi fizic sau online. Supraentitatea conține două subentități care categorizează modul de realizare a unei comenzi: CLIENT_FIZIC și CLIENT_ONLINE, având attribute specifice vânzării de produse fizic, respectiv online.

Cheia primară este reprezentată prin id_client.

Este o entitate independentă.

3) CLIENT_FIZIC

Un client care realizează comenzi fizic și căruia îi este specificat numărul de telefon. Subentitatea a lui CLIENT, deci dependentă de CLIENT.

Cheia primară este reprezentată prin id_client.

4) CLIENT_ONLINE

Un client care realizează comenzi online și căruia îi sunt specificate email-ul, numele și prenumele. Subentitatea a lui CLIENT, deci dependentă de CLIENT.

Cheia primară este reprezentată prin id_client.

5) COMANDA

O comandă este efectuată prin solicitarea unui client de a cumpăra piese de mobilier. Cantitățile specificate corespunzătoare se regăsesc la nivelul tabelului asociativ dintre COMANDA și PIESA_MOBILIER.

Comanda are asociată o adresă, dacă clientul optează pentru livrarea acesteia, entitatea ADRESA având participare optională.

Cheia primară este reprezentată prin id_comanda.

Este o entitate dependentă de ADRESA și AGENT_VANZARI.

Reguli relații:

- *Cantitatea dorită dintr-un produs la nivelul unei comenzi se reflectă la nivelul unui tabel asociativ dintre COMANDA și PIESA_MOBILIER.*

6) TRANZACTIE

O tranzacție implică transferul unei sume specifice de bani, modalitatea de plată presupunând și un indicator de confirmare a efectuării plății sau primirii sumei. Statusul plății

este de patru tipuri: „în procesare”, „verificare”, „aprobată”, „respinsă”, indiferent de tipul plășii (cash sau card).

Cheia primară este reprezentată prin id_tranzactie.

Este o entitate dependentă de COMANDA.

7) ADRESA

O adresă specifică localizarea unor clădiri ale firmei sau a locuințelor clienților care optează pentru livrarea comenzi.

Cheia primară este reprezentată prin id_adresa.

Este o entitate independentă.

Reguli relații:

- Adresa corespunde datelor unei locații ale entitășilor MAGAZIN sau COMANDA, însemnând că are participare optională raportat la ambele.

8) MAGAZIN

Magazinul reprezintă una dintre clădirile companiei de mobilier, aflată la una dintre locașile specifice, unde se desfășoară activitatea comercială și la care lucrează unul sau mai mulți angajați.

Cheia primară este reprezentată prin id_magazin.

Este o entitate independentă.

Reguli relații:

- Într-un magazin se desfășoară activitatea firmei, fiindu-i asociată o adresă.
- Pentru a indica disponibilitatea pieselor de mobilier, există o relașie de tip trei între MAGAZIN, STOC și PIESA_MOBILIER. De aceea, relașia stabilită va avea un tabel asociativ prin care vor fi sugerate detaliile de aprovizionare ale unui magazin, căruia îi sunt asociate mai multe stocuri, cu anumite produse, existând date exacte despre disponibilitatea acestora în stocul comun mai multor magazine.

9) STOC

Un stoc/inventar reține informașii referitor la actualizarea aprovizionării produselor și a capacitașii ocupate cu fiecare aprovizionare la nivelul tuturor produselor.

Cheia primară este o cheie compusă alcătuită din id_stoc și ultima_aprovizionare (data ultimei aprovizionări în vigoare la momentul realizării unei comenzi sau analizei logisticii).

Este o entitate independentă.

Reguli relații:

- Un stoc va avea o capacitate totală ocupată, ce reprezintă suma tuturor cantitășilor produselor din fiecare tip de mobilier de la nivelul stocului respectiv, iar cantitatea se regăsește în tabelul asociativ dintre STOC și PIESA_MOBILIER.

10) PIESA_MOBILIER

Piesa de mobilier este produsul comercializat de firma, însemnând că poate fi cumpărat, iar evidenșa lui este păstrată la nivelul unuia sau mai multor stocuri din magazinele în care mobilierul este vândut.

Cheia primară este reprezentată prin id_produs.

Este o entitate dependentă de CATEGORIE.

11) CATEGORIE

O categorie sugerează o clasificare a pieselor de mobilier în funcție de utilitatea lor în cadrul unui apartament.

Cheia primară este reprezentată prin id_categoria.

Este o entitate independentă.

12) OFERTA

O ofertă sugerează o reducere care se aplică unui produs de mobilier prin intermediul unui discount în timpul unei perioade stabilite de timp.

Cheia primară este o cheie compusă alcătuită din id_produs (cheie străină provenită din PIESA_MOBILIER) și data_inceput (data la care a intrat în aplicare discount-ul). De menționat că nu vor exista oferte la același produs care să înceapă de pe aceeași data succesiv.

Este o entitate dependentă de PIESA_MOBILIER, cheia străină fiind conținută chiar în cheia primară.

13) MATERIE_PRIMA

Materia primă este un material utilizat în producția pieselor de mobilier, fiind aduse de către un furnizor care are un contract de colaborare cu firma.

Fiecare materie primă diferă prin tipul materialului, motiv pentru care vor exista diferite unități de măsură și prețuri stabilite la o singură unitate.

Cheia primară este reprezentată prin id_material.

Este o entitate dependentă de FURNIZOR.

Reguli relații:

- Tabelul asociativ dintre PIESA_MOBILIER și MATERIE_PRIMA va include cantitatea exprimată printr-un *număr de unități dintr-un material* necesar în producția unui produs.

14) FURNIZOR

Un furnizor este o companie care aprovizează firma cu materiale de producție pentru realizarea pieselor de mobilier.

Cheia primară este reprezentată prin id_furnizor.

Este o entitate independentă.

Relații

○ AGENT_VANZARI—1(0)—M(0)— COMANDA

- Un agent de vânzări procesează mai multe comenzi sau niciuna.
- Fiecare comandă poate să fie validată de către un singur agent sau de niciunul.

Explicație:

- O comandă poate să nu fie verificată de către un agent numai în cazul în care clientul este de subtipul client_online, însemnând că o comandă a fost plasată în regim online de către client, iar acesta nu a solicitat un agent de vânzări.

Cardinalitate maximă: 1 : M

Cardinalitate minimă: 0 : 0

○ AGENT_VANZARI —M(0)—1(0)—MAGAZIN

- Un agent de vânzări lucrează la cel mult un magazin sau niciunul.

Explicație:

- Agentul poate să nu lucreze la niciun magazin, adică să nu se afle în evidență angajaților de la un magazin, în cazul în care procesează comenzi în regimul de lucru online.
- Un magazin are mai mulți agenti de vânzări sau niciunul.

Explicație:

- Pentru un magazin este funcțional și fără cel puțin un angajat care să proceseze comenzi.

Cardinalitate maximă: M : 1

Cardinalitate minimă: 0 : 0

○ CLIENT—1—M(1)—COMANDA

- Un client plasează una sau mai multe comenzi.

Explicație:

- Clientul trebuie să plaseze minim o comandă pentru a fi considerat un client al firmei.
- O comandă este asociată exact unui client.

Cardinalitate maximă: 1 : M

Cardinalitate minimă: 1 : 1

○ COMANDA—1—1—TRANZACTIE

- Unei comenzi îi corespunde exact o tranzacție.
- O tranzacție reiese din exact o comandă.

Cardinalitate maximă: 1 : 1

Cardinalitate minimă: 1 : 1

○ COMANDA—M(0)—M(1)—PIESA_MOBILIER

- O comandă conține una sau mai multe piese de mobilier.
- O piesă de mobilier e inclusă în mai multe comenzi sau niciuna.

Cardinalitate maximă: M : M

Cardinalitate minimă: 0 : 1

○ COMANDA—M(0)—1(0)—ADRESA

- O comandă poate fi livrată la cel mult o adresă sau niciuna.

Explicație:

- Comanda nu este livrată la nicio adresă în cazul în care clientul nu optează pentru livrarea acesteia.

- O adresă este asociată mai multor comenzi sau niciuneia.

Explicație:

- Adresa se află în legătură și cu un magazin, însemnând că este adresa unei locații a unui magazin sau adresa locației de livrare a comenzi unui client.

Cardinalitate maximă: M : 1

Cardinalitate minimă: 0 : 0

○ MAGAZIN—1(0)—1—ADRESA

- Un magazin se află la exact o adresă/ locație.

Explicație:

- Un magazin IKEA poate să fie doar un punct de ridicare a produselor, adresa lui servind în multe cazuri drept adresă de livrare a comenzilor clientilor.

- O adresă aparține cel mult unui magazin sau niciunui.

Explicație:

- Adresa se află în legătură și cu o comandă, însemnând că este adresa unei locații a unui magazin sau adresa locației de livrare a comenzii unui client.

Cardinalitate maximă: 1 : 1

Cardinalitate minimă: 0 : 1

○ MAGAZIN ——STOC——PIESA_MOBILIER

Relație de tip trei care leagă entitățile MAGAZIN, STOC și PIESA_MOBILIER.

- Un magazin are acces la mai multe stocuri sau niciunul și vinde mai multe piese de mobilier sau niciuna, regăsite în stocuri.
- Un stoc e asociat mai multor magazine sau niciunui. Un stoc conține mai multe piese de mobilier sau niciuna. Un magazin are asociat mai multe stocuri sau niciunul. Un magazin vinde mai multe piese de mobilier sau niciuna. O piesă de mobilier se află în mai multe stocuri sau niciunul. O piesă de mobilier e vândută la mai multe magazine sau niciunul.

Cardinalitate maximă în dreptul fiecărei entități: M

Cardinalitate minimă în dreptul fiecărei entități: 0

○ PIESA_MOBILIER—M(0)—1—CATEGORIE

- O piesă de mobilier aparține exact unei categorii.
- O categorie are mai multe piese de mobilier sau niciuna.

Cardinalitate maximă: M : 1

Cardinalitate minimă: 0 : 1

○ PIESA_MOBILIER—1—M(0)—OFERTA

- O piesă de mobilier are mai multe oferte sau niciuna.
- O ofertă cuprinde exact o piesă de mobilier.

Cardinalitate maximă: 1 : M

Cardinalitate minimă: 1 : 0

○ PIESA_MOBILIER—M(1)—M(0)—MATERIE_PRIMA

- O piesă de mobilier este produsă mai multe materii prime sau niciuna.

Explicație:

- Nu orice piesă de mobilier este produsă de către firma, nu întotdeauna fiind cunoscute materialele, alături de cantitățile folosite și prețul per unitate, deoarece unele produse provin de la terți. Această alegere nu respectă modelul real al magazinului IKEA.

- O materie primă este folosită la producția uneia sau mai multor piese de mobilier.

Explicație:

- Dacă cardinalitatea minimă ar fi fost zero, nu ar fi fost justificată aprovisionarea cu acel tip de material.

Cardinalitate maximă: M : M

Cardinalitate minimă: 1 : 0

- MATERIE_PRIMA—M(1)—1—FURNIZOR
 - O materie primă este aprovisionată de către exact un furnizor.
 - Un furnizor aprovisionează una sau mai multe materii prime.

Cardinalitate maximă: M : 1

Cardinalitate minimă: 1 : 1

Atribute

- 1) AGENT_VANZARI
 - id_angajat number(5) primary key,
 - cod unic, de tip numeric, de exact 5 cifre
 - nume varchar(20) not null,
 - de tip caracter de lungime maximă 20, diferă de null
 - prenume varchar(20) not null,
 - de tip caracter de lungime maximă 20, diferă de null
 - data_angajare date not null,
 - de tip data calendaristică, diferă de null
 - telefon char(12) unique,
 - de tip caracter, de exact 12 cifre, unic
- 2) CLIENT
 - id_client number(8) primary key,
 - cod unic, de tip numeric, de exact 8 cifre
 - tip_client varchar(6) not null, constraint verif_tip_client check (tip_client in ('online', 'fizic')),
 - de tip caracter de lungime maximă 6, diferă de null, are drept constrângere una dintre valorile „online” sau „fizic”
 - nume_firma varchar(50)
 - de tip caracter de lungime maximă 50
- 3) CLIENT_FIZIC
 - telefon char(12) unique,
 - de tip caracter, de exact lungime 12, unic
- 4) CLIENT_ONLINE
 - email varchar(50) not null,
 - de tip caracter de lungime maximă 50, diferă de null
 - nume varchar(20),
 - de tip caracter de lungime maximă 20
 - prenume varchar(20) not null,
 - de tip caracter de lungime maximă 20, diferă de null
- 5) COMANDA
 - id_comanda number(10) primary key,

- cod unic, de tip numeric, de exact 10 cifre
- pret number(6, 2) not null,
 - de tip numeric, având în total 6 cifre, dintre care ultimele 2 sunt zecimale, diferă de null
- data_achizitie date default sysdate,
 - de tip dată calendaristică, valoare implicită sysdate

6) TRANZACTIE

- id_tranzactie number(10) primary key,
 - cod unic, de tip numeric, de exact 10 cifre
- modalitate_plata char(4) not null, constraint verif_modal_plata check (modalitate_plata = 'cash' or modalitate_plata = 'card'),
 - de tip caracter, de exact lungime 4, având drept constrângere una dintre valorile „cash” sau „card”
- status_plata varchar(12) default 'aprobata', constraint verif_status check (status_plata in ('in procesare', 'verificare', 'aprobata', 'respsina')),
 - de tip caracter, de lungime maximă 12, având drept constrângere una dintre valorile „in procesare”, „verificare”, „aprobata”, „respsina”

7) ADRESA

- cod_postal number(6) primary key,
 - cod unic, de tip numeric, de 6 cifre
- oras varchar(25) not null,
 - de tip caracter, de lungime maximă de 25 de caractere, diferă de null
- tara varchar(25) not null,
 - de tip caracter, de lungime maximă de 25 de caractere, diferă de null
- strada varchar(70)
 - de tip caracter, de lungime maximă de 70 de caractere

8) MAGAZIN

- id_magazin varchar(4) primary key,
 - cod unic, de tip caracter, de lungime maximă de 4 caractere
- telefon char(12) unique,
 - de tip caracter, de exact 12 cifre, unic
- centru_ridicare smallint not null, constraint verif_centru_bool check(centru_ridicare in (0, 1)),
 - de tip întreg, diferă de null, având drept constrângere una dintre valorile 0 sau 1

9) STOC

- id_stoc char(5),
 - cod unic, de tip caracter, de lungime exact 5
- data_aprovizionare date default sysdate,
 - de tip dată calendaristică, valoare implicită sysdate

10) PIESE_MOBILIER

- id_produs number(6) primary key,
 - cod unic, de tip numeric, de exact 6 cifre
- nume varchar(30) unique,
 - de tip caracter, de lungime maximă 30, unic
- pret number(6, 2) not null,

- de tip numeric, având în total 6 cifre, dintre care ultimele 2 sunt zecimale, diferă de null
- descriere varchar(100),
 - de tip caracter, de lungime maximă 100
- lungime int,
 - de tip întreg
- latime int,
 - de tip întreg
- inaltime int,
 - de tip întreg
- garantie varchar(50),
 - de tip caracter, de lungime maximă 50
- link_web varchar(150),
 - de tip caracter, de lungime maximă 150
- nume_designer varchar(50),
 - de tip caracter, de lungime maximă 50

11) CATEGORIE

- id_categorie char(5) primary key,
 - cod unic, de tip caracter, de exact lungime de 5
- nume varchar(50) unique
 - de tip caracter, de lungime maximă 50, unic

12) OFERTA

- id_produs number(6),
 - cod unic, de tip numeric, de exact 6 cifre
- data_inceput date default sysdate,
 - de tip dată calendaristică, valoare implicită sysdate
- data_sfarsit date not null,
 - de tip dată calendaristică, diferă de null
- discount number(5, 2) not null, constraint verif_procent check(discount > 0 and discount < 100),
 - de tip numeric, având în total 5 cifre, dintre care ultimele 2 sunt zecimale, diferă de null, având drept constrângere o valoare între 0 și 100

13) MATERIE_PRIMA

- id_material number(10) primary key,
 - cod unic, de tip numeric, de exact 10 cifre
- tip_material varchar(25) not null,
 - de tip caracter, de lungime maximă 25, diferă de null
- unitate_masura varchar(17) not null, constraint verif_masura check (unitate_masura in ('metru patrat', 'decimetru patrat', 'centimetru patrat', 'milimetru patrat', 'metru cub', 'decimetru cub', 'centimetru cub', 'milimetru cub', 'kilogram', 'gram')),
 - de tip caracter, de lungime maximă 17, diferă de null, având drept constrângere o unitate de măsură validă
- pret_unitate decimal not null,
 - de tip decimal, diferă de null

14) FURNIZOR

- id_furnizor varchar(4) primary key,
 - cod unic, de tip caracter, de lungime maximă 4

- nume varchar(35) unique,
 - de tip caracter, de lungime maximă 35, unic
- telefon char(12) unique
 - de tip caracter, de lungime maximă 12, unic

15) ADAUGA_COMANDA

- cantitate int default 1,
 - de tip întreg, valoare implicită 1
- moment_timp timestamp default systimestamp,
 - de tip moment de timp, valoare implicită sysdate

16) PRODUSA_DIN

- unitati int,
 - de tip întreg

17) APROVIZIONEaza

- cantitate int,
 - de tip întreg

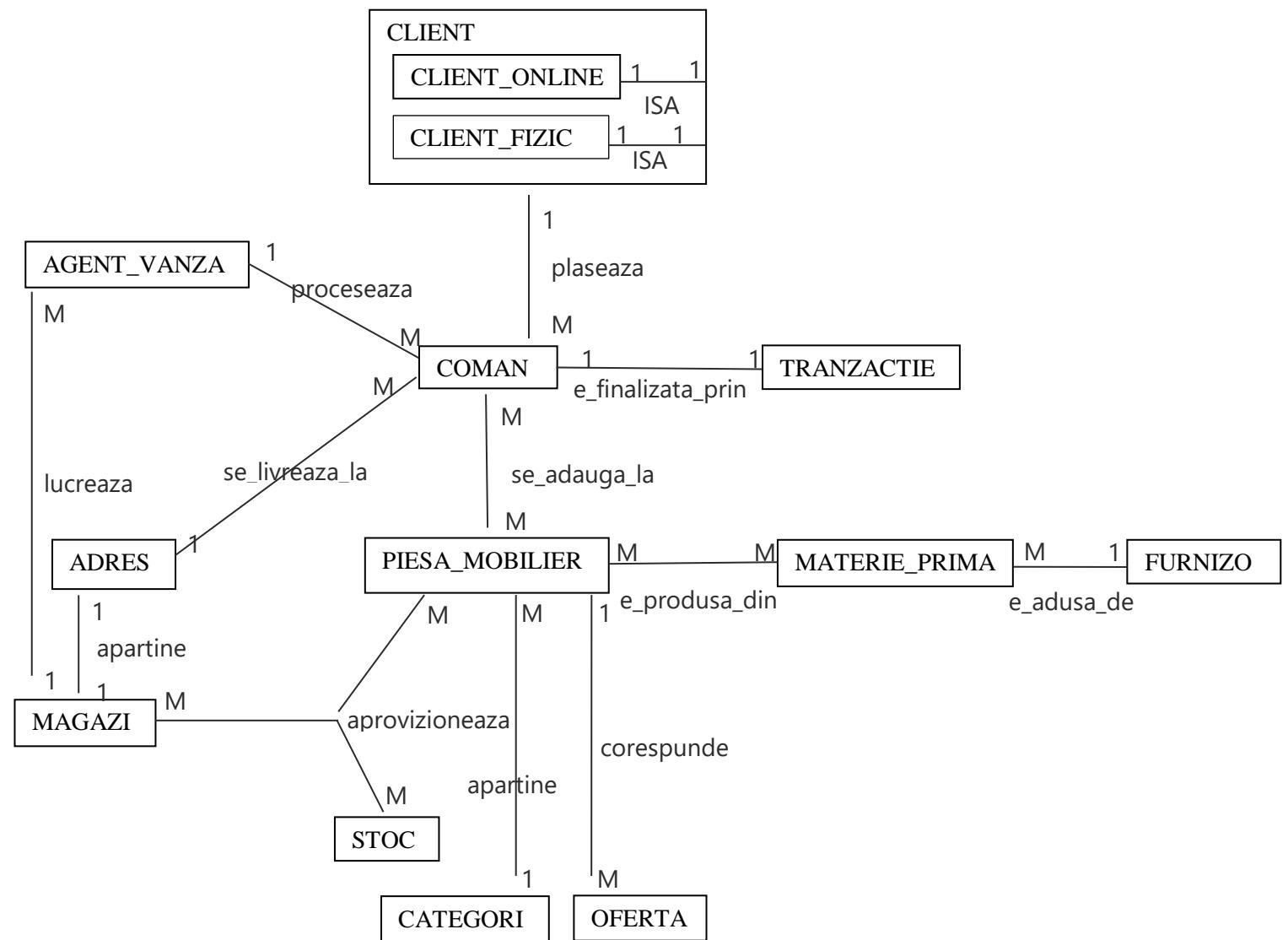


Diagrama entitate-relație

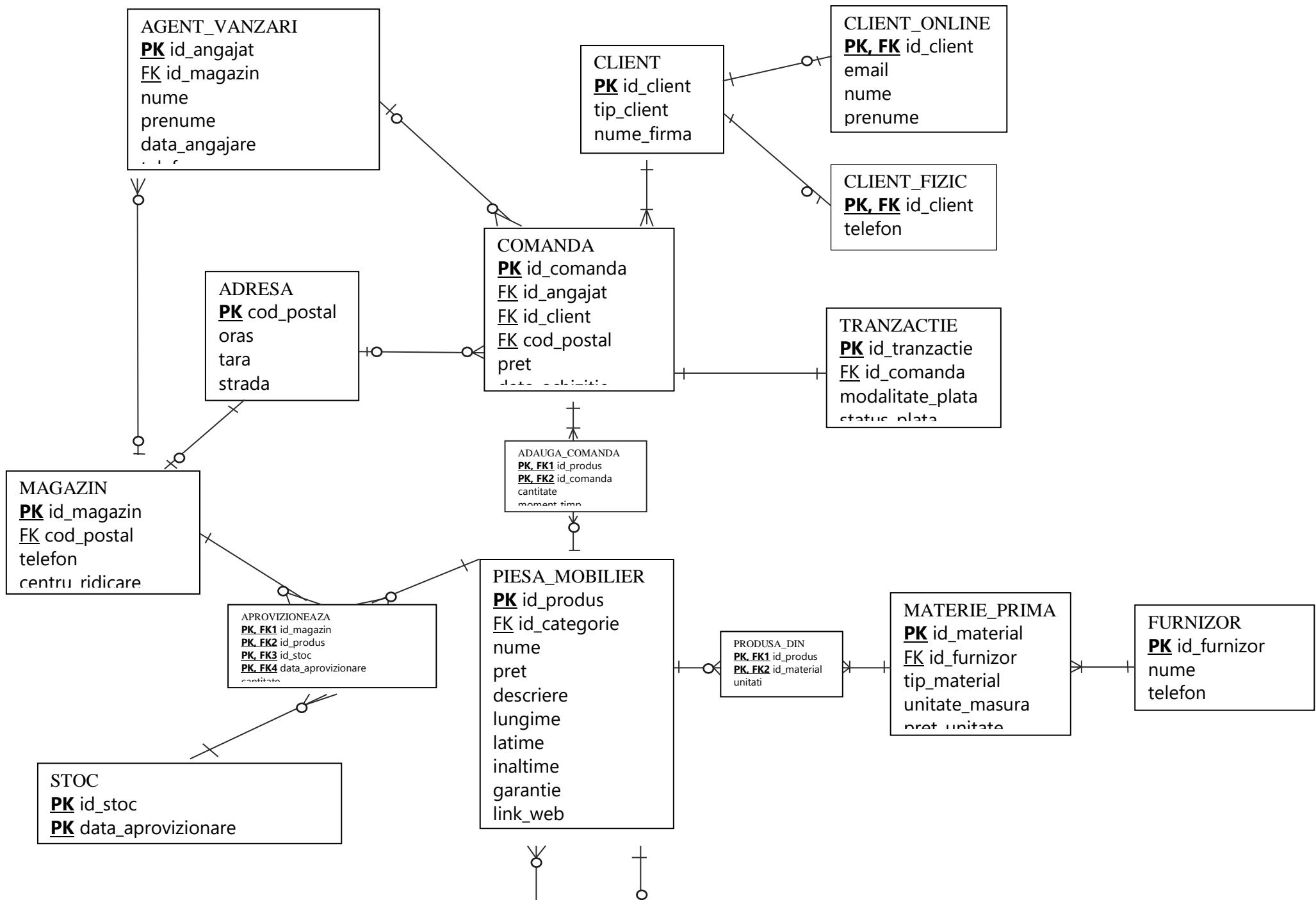
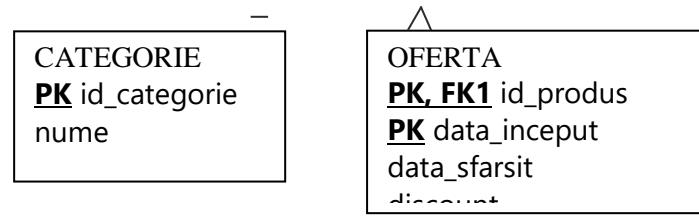


Diagrama conceptuală



Schemele relationale corespunzătoare diagramei conceptuale

AGENT_VANZARI(id_angajat#, nume, prenume, data_angajare, telefon, id_magazin)

CLIENT(id_client#, tip_client, nume_firma)

CLIENT_ONLINE(id_client#, tip_client, nume_firma, email, nume, prenume)

CLIENT_FIZIC(id_client#, tip_client, nume_firma, telefon)

TRANZACTIE(id_tranzactie#, modalitate_plata, status_plata, id_comanda)

ADRESA(cod_postal#, oras, tara, strada)

COMANDA(id_comanda#, pret, data_achizitie, id_angajat, id_client, cod_postal)

MAGAZIN(id_magazin#, telefon, centru_ridicare, cod_postal)

STOC(id_stoc#, ultima_aprovizionare#)

PIESA_MOBILIER(id_produs#, nume, pret, descriere, lungime, latime, inaltime, garantie, link_web, nume_designer, id_categorie)

CATEGORIE(id_categorie#, nume)

OFERTA(id_produs#, data_inceput#, data_sfarsit, discount)

MATERIE_PRIMA(id_material#, tip_material, unitate_masura, pret_unitate, id_furnizor)

FURNIZOR(id_furnizor#, nume, telefon)

ADAUGA_COMANDA(id_produs#, id_comanda#, cantitate, moment_timp)

PRODUSA_DIN(id_produs#, id_material#, unitati)

APROVIZIONEAZA(id_magazin#, id_produs#, id_stoc#, data_aprovizionare#, cantitate)

Realizarea normalizării până la FN3

I. FN1

O relație se prezintă în FN1 dacă valorile fiecărui câmp sunt atomice și fiecare tuplu este unic, însemnând că orice combinație de coloane formează un rând unic de fiecare dată, iar fiecare înregistrare poate fi identificată printr-o cheie primară. Baza de date respectă criteriile FN1, deoarece orice înregistrare la nivelul fiecărei entități definită este identificată prin intermediul cheilor primare.

Un exemplu de non-FN1 ar fi dacă entitatea AGENT_VANZARI ar avea atributele nume, prenume și data_angajare, ar însemna că nu se aplică caracteristicile identificării unice a unui angajat, deoarece ar putea să existe aceeași combinație de date, însă să reprezinte un alt rând sau înregistrare în baza de date. În acest caz, există posibilitatea existenței a cel puțin doi agenți de vânzări care să aibă același nume, prenume și aceeași dată a angajării. Pentru a transforma acest exemplu în FN1, trebuie adăugată cheia primară id_angajat.

II. FN2

O relație se prezintă în FN2, dacă se află în FN1 și fiecare atribut care nu este o cheie să fie dependent funcțional de cheia primară, fără a exista dependințe parțiale. Baza de date

respectă criteriile FN2, deoarece fiecare atribut al oricărei entități depinde total de cheia primară corespunzătoare.

A existat o greșală de proiectare a bazei de date, în stadiile de început, la nivelul entității STOC, care în plus conținea atributul capacitate_maxima. Cheia primară o reprezintă tuplul (id_magazin, secțiune_magazin, ultima_aprovizionare), acest ultim atribut ce indică data ultimei aprovizionări ar fi însemnat că orice înregistrare a stocului pentru care primele două componente ale cheii sunt identice să fie o înregistrare care să conțină mereu aceeași valoare a capacitatei maxime, deoarece în mod general capacitatea totală a unui inventar nu este variabilă. În acest caz, capacitate_maxima depinde numai de (id_magazin, secțiune_magazin), în loc să depindă de întreaga cheie.

Un exemplu de non-FN2 ar fi dacă entitatea PIESA_MOBILIER ar avea, în plus față de attributele definite id_produs#, nume, pret, descriere, lungime, latime, inaltime, garantie, link, id_categorie, unul sau mai multe attribute de tipul material_1, material_2, ..., ce ar reprezenta materiile prime din care ar fi produsă o piesă. Această schimbare ar însemna să existe suficiente attribute pentru a putea descrie chiar și piesa de mobilier în componență căreia există cele mai multe materiale. Se observă că noile attribute nu depind total de cheia primară id_produs, deoarece există posibilitatea ca mai multe piese de mobilier să fie compuse din aceleași materiale, astfel apărând valori multiple care s-ar repeta. Pentru a transforma acest exemplu în FN2, trebuie eliminate attributele menționate, fiind generată o nouă entitate numită MATERIE_PRIMA și stabilindu-se o relație de many to many la capătul entității MATERIE_PRIMA.

III. FN3

O relație se prezintă în FN3, dacă se află în FN2 și nu există dependință funcțională tranzitivă, însemnând că fiecare atribut care nu este o cheie să fie dependent de un alt atribut care nu e cheie, care la rândul său să fie dependent de cheia relației. Baza de date respectă criteriile FN3.

Au existat două greșeli de proiectare a bazei de date, în stadiile de început.

Prima greșală se referă la nivelul supraentității CLIENT, care avea ca attribute id_client#, tip_client, tip_legal, nume_firma și CUI_firma. Se observă cum nume_firma și CUI_firma depind de tip_legal care ar fi avut ca valori posibile „persoana fizica” sau „persoana juridica”, însemnând că o doar pentru o persoană juridică ar fi fost completate date în legătură cu firma reprezentată. O soluție, care s-a dovedit a fi greșită, a constat în eliminarea atributului tip_legal, cu toate că ar fi verificat funcționalitatea, prin intermediul căruia este respectată atât convenția menționată, cât și verificată existența succesivă a unor date în ambele câmpuri. De aceea, soluția propusă nu garantează dacă atunci când nume_firma e completat, CUI_firma nu ar rămâne necompletat, însemnând că dependența tranzitivă rămâne. Așadar, alegerea a fost să elimin tip_legal și CUI_firma, fiind de data aceasta respectate criteriile FN3.

În timp ce a două greșală se referă la entitatea COMANDA, în care există și atributul livrare, care ar fi reprezentat valorile 'da' sau 'nu' dacă clientul ar fi optat pentru livrarea produselor achiziționate, însemnând că cod_postal, cheia străină provenită din ADRESA ar fi depins de această opțiune [constraint verif_asociere_adresa check((livrare = 'da' and cod_postal is not null) or (livrare = 'nu' and cod_postal is null))].

Un exemplu de non-FN3 ar fi dacă entitatea COMANDA ar avea id_comanda#, pret, data_achizitie, livrare, cod_postal, adresa, similar ar fi și pentru entitatea MAGAZIN care ar avea atributele: id_magazin#, telefon, cod_postal, adresa. Se observă cum adresa, ce reprezintă o înlățuire care specifică orașul, țara și strada, nu depinde direct de id_comanda sau id_magazin, ci de cod_postal, care la rândul sau depinde de cheile primare menționate în ambele cazuri. Pentru a transforma acest exemplu în FN3, se elimină atributul adresa, care nu depinde direct de cheile primare și este creată o nouă entitate ADRESA cu atributele cod_postal#, oras, tara, strada, iar cheia din ADRESA devine o cheie străină în COMANDA, respectiv MAGAZIN.

Proiectare stadiu de început abandonată

- Pentru a indica disponibilitatea pieselor de mobilier, există relații directe între MAGAZIN și STOC și cea dintre STOC și PIEZA_MOBILIER, însă participarea optională a stocului obligă rezolvarea unei capcane de întrerupere, cu scopul de a putea răspunde la interogări între MAGAZIN și PIEZA_MOBILIER. De aceea, relația stabilită va avea un tabel asociativ prin care vor fi sugerate numai regulile de aprovizionare ale unui magazin cu anumite produse, fără să existe date exacte despre disponibilitatea acestora în stoc (acest fapt e vizibil numai la nivelul tabelului asociativ stabilit între STOC și PIEZA_MOBILIER).

Reconfigure project

```
drop sequence seq_client;
drop sequence seq_agent;
drop sequence seq_comanda;
drop sequence seq_tranzactie;
drop sequence seq_material;
drop sequence seq_produs;

drop table adauga_comanda cascade constraints;
drop table piesa_mobilier cascade constraints;
drop table categorie cascade constraints;
drop table comanda cascade constraints;
drop table adresa cascade constraints;
drop table agent_vanzari cascade constraints;
drop table aprovizioneaza cascade constraints;
drop table produsa_din cascade constraints;
drop table client cascade constraints;
drop table client_fizic cascade constraints;
drop table client_online cascade constraints;
drop table furnizor cascade constraints;
drop table magazin cascade constraints;
drop table materie_prima cascade constraints;
drop table stoc cascade constraints;
drop table tranzactie cascade constraints;
drop table oferta cascade constraints;
drop table detalii_produse;
```

The screenshot shows the Oracle SQL Developer interface with two main panes. The left pane displays the database structure under 'Tables (Filtered)' for the schema 'SGBD_HOMEDB1-Alexandrescu'. The right pane shows the 'Script Output' window with the results of a script execution.

```

drop table client cascade constraints;
drop table client_fizic cascade constraints;
drop table client_online cascade constraints;
drop table furnizor cascade constraints;
drop table magazin cascade constraints;
drop table materie_prima cascade constraints;
drop table stoc cascade constraints;
drop table tranzactie cascade constraints;

Sequence SEQ_CLIENT dropped.

Sequence SEQ_AGENT dropped.

Sequence SEQ_COMANDA dropped.

Sequence SEQ_TRANZACTIE dropped.

Sequence SEQ_MATERIAL dropped.

Sequence SEQ_PRODUS dropped.

Table ADAUGA_COMANDA dropped.

Table PIESA_MOBILIER dropped.

Table CATEGORIE dropped.

Table COMANDA dropped.

Table ADRESA dropped.

Table AGENT_VANZARI dropped.

Table APROVIZIONEAZA dropped.

Table PRODUSA_DIN dropped.

Table CLIENT dropped.

Table CLIENT_FIZIC dropped.

Table CLIENT_ONLINE dropped.

Table FURNIZOR dropped.

Table MAGAZIN dropped.

Table MATERIE_PRIMA dropped.

Table STOC dropped.

Table TRANZACTIE dropped.

Table OFERTA dropped.

Table DETALII_PRODUSE dropped.

```

Secvențe pentru inserarea înregistrărilor în tabele

```

create sequence seq_client
start with 10000000
increment by 10
maxvalue 99999999
nocycle
nocache;

```

```

create sequence seq_agent
start with 10000
increment by 5

```

```
maxvalue 99999
nocycle
nocache;

create sequence seq_comanda
start with 1000000000
increment by 2
maxvalue 4999999999
nocycle
nocache;

create sequence seq_tranzactie
start with 1000000001
increment by 2
maxvalue 4999999999
nocycle
nocache;

create sequence seq_material
start with 5000000000
increment by 3
maxvalue 9999999999
nocycle
nocache;

create sequence seq_produs
start with 400000
increment by 1
maxvalue 999999
nocycle
nocache;

/*select seq_client.nextval
from dual;
select seq_agent.nextval
from dual;
select seq_comanda.nextval
from dual;
select seq_tranzactie.nextval
from dual;
select seq_produs.nextval
from dual;
select seq_material.nextval
from dual;*/
```

The screenshot shows the Oracle SQL Developer interface. On the left, the database structure is displayed under 'Tables (Filtered)', showing tables like ADAUGA_COMANDA, ADRESA, AGENT_VANZARI, APROVIZIONEAZA, CATEGORIE, CLIENT, CLIENT_FIZIC, and CLIENT_ONLINE. Below this is a 'Reports' section with options like All Reports, Analytic View Reports, Data Dictionary Reports, etc. In the center, a script editor window contains the following PL/SQL code:

```

INCREMENT BY 1
maxvalue 9999999999
nocycle
nocache;

create sequence seq_produs
start with 400000

```

Below the code, the 'Script Output' window shows the results of the execution:

```

Sequence SEQ_CLIENT created.

Sequence SEQ_AGENT created.

Sequence SEQ_COMANDA created.

Sequence SEQ_TRANZACTIE created.

Sequence SEQ_MATERIAL created.

Sequence SEQ_PRODUS created.

```

Crearea tabelelor și inserarea datelor

Modificare datetime și timestamp format

```

alter session set nls_date_format='DD-MON-RR';
alter session set nls_timestamp_format='DD-MON-RR HH24:MI:SS';
select systimestamp from dual;

```

The screenshot shows the Oracle SQL Developer interface. On the left, the database structure is displayed under 'Tables (Filtered)'. In the center, a script editor window contains the following PL/SQL code:

```

constraint fk_aprovizioneaza_id_stoc_data_aprovizionare
foreign key(id_stoc, data_aprovizionare)
references STOC(id_stoc, data_aprovizionare)
on delete cascade
);

alter session set nls_date_format='DD-MON-RR';
alter session set nls_timestamp_format='DD-MON-RR HH24:MI:SS';
select systimestamp from dual;

insert into CLIENT values(seq_client.nextval, 'online', null);
insert into CLIENT values(seq_client.nextval, 'fizic', 'DRLuxx_Store')

```

Below the code, the 'Script Output' window shows the results of the execution:

```

Session altered.

Session altered.

>>Query Run In:Query Result

```

SYSTIMESTAMP	
1 01-JAN-25 08.04.09.748000000 PM +02:00	

Creare tabele

```
create table CLIENT
(
    id_client number(8) primary key,
    tip_client varchar(6) not null, constraint verif_tip_client check
(tip_client in ('online', 'fizic')),
    nume_firma varchar(50)
);

create table CLIENT_ONLINE
(
    id_client number(8) primary key, -- id_client_online
    email varchar(50) not null,
    nume varchar(20),
    prenume varchar(20) not null,
    constraint fk_client_online
        foreign key(id_client)
        references CLIENT(id_client)
        on delete cascade
);

create table CLIENT_FIZIC
(
    id_client number(8) primary key, -- id_client_fizic
    telefon char(12) unique,
    constraint fk_client_fizic
        foreign key(id_client)
        references CLIENT(id_client)
        on delete cascade
);

create table ADRESA
(
    cod_postal number(6) primary key,
    oras varchar(25) not null,
    tara varchar(25) not null,
    strada varchar(70)
);

create table MAGAZIN
(
    id_magazin varchar(4) primary key,
    telefon char(12) unique,
    centru_ridicare smallint, constraint verif_centru_bool
check(centru_ridicare in (0, 1)),
```

```

cod_postal number(6) not null unique, -- relatie one-to-one: cardinalitate
maxima 1 (unique) inspre MAGAZIN, cardinalitate minima 1 (not null) inspre
ADRESA

constraint fk_magazin_cod_postal
  foreign key(cod_postal)
  references ADRESA(cod_postal)
  on delete set null
);

create table AGENT_VANZARI
(
  id_angajat number(5) primary key,
  nume varchar(20) not null,
  prenume varchar(20) not null,
  data_angajare date not null,
  telefon char(12) unique,
  id_magazin varchar(4), -- cardinalitate maxima M (not unique) inspre
  AGENT_VANZARI, cardinalitate minima 0 (may be null) inspre MAGAZIN

  -- imposibil fara un trigger: constraint verif_tip_magazin
  check(centru_ridicare = 0),
  constraint fk_agent_vanzari_id_magazin
    foreign key(id_magazin)
    references MAGAZIN(id_magazin)
    on delete set null -- raman angajati dar nu le mai este asociat
    magazinul cu inregistrarea stearsa
);

create table COMANDA
(
  id_comanda number(10) primary key,
  pret number(6, 2) not null,
  data_achizitie date default sysdate,
  --livrare char(2) default 'nu', constraint verif_opt_livrare check(livrare
  in ('da', 'nu')),
  id_angajat number(5), -- cardinalitate minima 0 inspre ANGAJAT
  id_client number(8) not null, -- cardinalitate minima 1 inspre CLIENT
  cod_postal number(6), -- cardinalitate minima 0 inspre ADRESA
  --constraint verif_asociere_adresa check((livrare = 'da' and cod_postal is
  not null) or (livrare = 'nu' and cod_postal is null)), -- este dependinta
  tranzitiva, nu respecta criteriile FN3

  -- relatii one-to-many (many in partea tablei COMANDA)
  constraint fk_comanda_id_angajat
    foreign key(id_angajat)
    references AGENT_VANZARI(id_angajat)
    on delete set null,

```

```

constraint fk_comanda_id_client
    foreign key(id_client)
    references CLIENT(id_client)
    on delete set null,
constraint fk_comanda_cod_postal
    foreign key(cod_postal)
    references ADRESA(cod_postal)
    on delete set null
);

create table TRANZACTIE
(
    id_tranzactie number(10) primary key,
    modalitate_plata char(4) not null, constraint verif_modal_plata check
(modalitate_plata = 'cash' or modalitate_plata = 'card'),
    status_plata varchar(12) default 'aprobata', constraint verif_status check
(status_plata in ('in procesare', 'verificare', 'aprobata', 'respinsa')), --
atat pentru o plata card, cat si cash, cu toate ca un model real nu respecta
aceasta regula
    id_comanda number(10) not null unique, -- relatie one-to-one:
cardinalitate maxima 1 (unique) inspre TRANZACTIE, cardinalitate minima 1 (not
null) inspre COMANDA

constraint fk_tranzactie_id_comanda
    foreign key(id_comanda)
    references COMANDA(id_comanda)
    on delete cascade
);

create table CATEGORIE
(
    id_categorie char(5) primary key,
    nume varchar(50) unique
);

create table PIESA_MOBILIER
(
    id_produs number(6) primary key,
    nume varchar(30) unique,
    pret number(6, 2) not null, -- o piesa de mobilier va avea intotdeauna un
pret listat si maximul unui pret nu depaseste pragul de 9999 de Lei
    descriere varchar(100),
    lungime int,
    latime int,
    inaltime int,
    garantie varchar(50),
    link_web varchar(150),
    nume_designer varchar(50),
);

```

```

    id_categorie char(5) not null, -- cardinalitate maxima M (not unique)
inspire PIESA_MOBILIER, cardinalitate minima 1 (not null) inspire CATEGORIE

constraint fk_piesa_mobilier_id_categorie
    foreign key(id_categorie)
    references CATEGORIE(id_categorie)
    on delete cascade
);

create table OFERTA
(
    id_produs number(6),
    data_inceput date default sysdate,
    data_sfarsit date not null,
    discount number(5, 2) not null, constraint verif_procent check(discount >
0 and discount < 100),

    constraint pk_oferta primary key(id_produs, data_inceput),
    constraint fk_oferta_id_produs
        foreign key(id_produs)
        references PIESA_MOBILIER(id_produs)
        on delete cascade
);

create table ADAUGA_COMANDA
(
    id_produs number(6),
    id_comanda number(10),
    cantitate int default 1,
    moment_timp timestamp default systimestamp,

    constraint pk_adauga_comanda primary key(id_produs, id_comanda),
    constraint fk_adauga_comanda_id_produs
        foreign key(id_produs)
        references PIESA_MOBILIER(id_produs)
        on delete cascade,
    constraint fk_adauga_comanda_id_comanda
        foreign key(id_comanda)
        references COMANDA(id_comanda)
        on delete cascade
);

create table FURNIZOR
(
    id_furnizor varchar(4) primary key,
    nume varchar(35) unique,
    telefon char(12) unique
);

```

```

create table MATERIE_PRIMA
(
    id_material number(10) primary key,
    tip_material varchar(25) not null, -- nu e unic deoarece mai multi
furnizori pot aproviziona cu acelasi tip de material, avand aceeasi denumire
    unitate_masura varchar(17) not null, constraint verif_masura check
(unitate_masura in ('metru patrat', 'decimetru patrat', 'centimetru patrat',
'milimetru patrat', 'metru cub', 'decimetru cub', 'centimetru cub', 'milimetru
cub', 'kilogram', 'gram')),
    pret_unitate decimal not null,
    id_furnizor varchar(4) not null, -- cardinalitate maxima M (not unique)
inspre MATERIE_PRIMA, cardinalitate minima 1 (not null) inspre FURNIZOR

    constraint fk_materie_prima_id_furnizor
        foreign key(id_furnizor)
        references FURNIZOR(id_furnizor)
        on delete cascade
);

create table PRODUSA_DIN
(
    id_produs number(6),
    id_material number(10),
    unitati int,

    constraint pk_produsa_din primary key(id_produs, id_material),
    constraint fk_produsa_din_id_produs
        foreign key(id_produs)
        references PIESA_MOBILIER(id_produs)
        on delete cascade,
    constraint fk_produsa_din_id_material
        foreign key(id_material)
        references MATERIE_PRIMA(id_material)
        on delete cascade
);

create table STOC
(
    id_stoc char(5),
    data_aprovizionare date default sysdate,

    constraint pk_stoc primary key(id_stoc, data_aprovizionare)
);

create table APROVIZIONEAZA
(
    id_magazin varchar(4),

```

```

        id_produs number(6),
        id_stoc char(5),
        data_aprovizionare date,
        cantitate int,

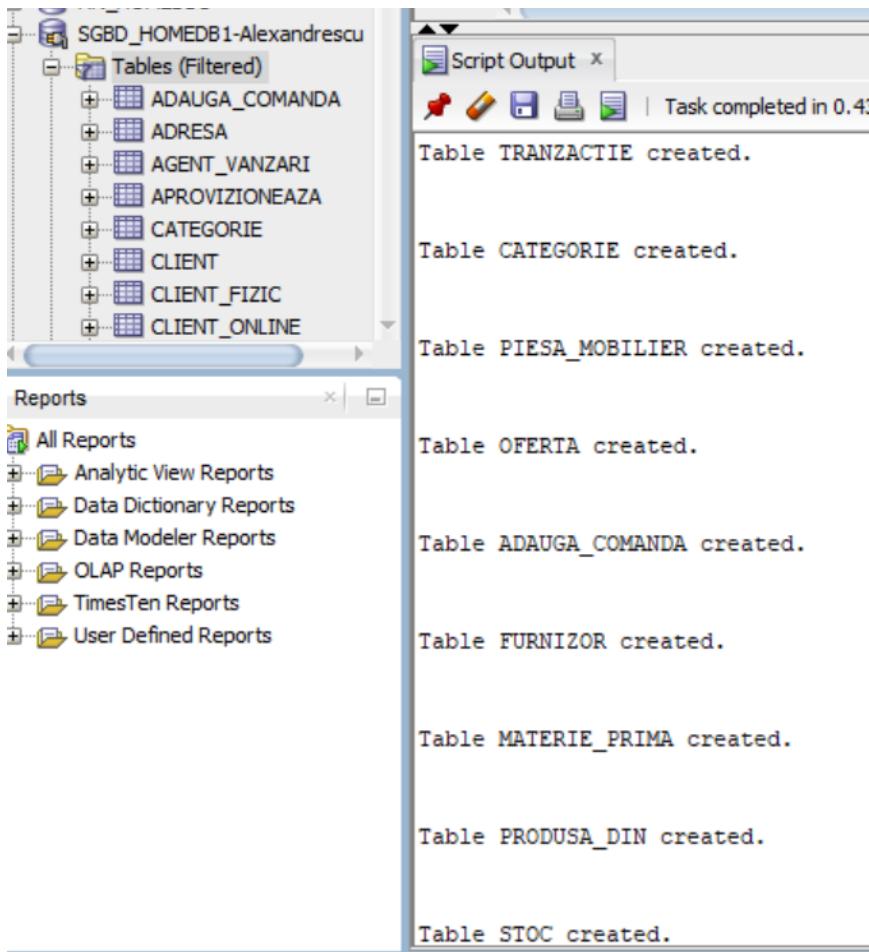
        constraint pk_aprovizioneaza primary key(id_magazin, id_produs, id_stoc,
data_aprovizionare),
        constraint fk_aprovizioneaza_id_produs
            foreign key(id_produs)
            references PIESA_MOBILIER(id_produs)
            on delete cascade,
        constraint fk_aprovizioneaza_id_magazin
            foreign key(id_magazin)
            references MAGAZIN(id_magazin)
            on delete cascade,
        constraint fk_aprovizioneaza_id_stoc_data_aprovizionare
            foreign key(id_stoc, data_aprovizionare)
            references STOC(id_stoc, data_aprovizionare)
            on delete cascade
    );

```

The screenshot shows the Oracle SQL Developer interface. On the left, the Database Navigator displays a list of tables under the schema 'SGBD_HOMEDB1-Alexandrescu'. The tables listed are: ADAUGA_COMANDA, ADRESA, AGENT_VANZARI, APROVIZIONEAZA, CATEGORIE, CLIENT, CLIENT_FIZIC, and CLIENT_ONLINE. On the right, the Script Output window shows the execution of a script that creates these tables. The output messages are:

- Table CLIENT created.
- Table CLIENT_ONLINE created.
- Table CLIENT_FIZIC created.
- Table ADRESA created.
- Table MAGAZIN created.
- Table AGENT_VANZARI created.
- Table COMANDA created.

The 'Script Output' window also indicates that the task completed successfully.



Inserare date

```
insert into CLIENT values(seq_client.nextval, 'online', null);
insert into CLIENT values(seq_client.nextval, 'fizic', 'DEluxe Store');
insert into CLIENT values(seq_client.nextval, 'online', 'Retail Express
S.R.L.');
insert into CLIENT values(seq_client.nextval, 'online', 'Alina CraftShop
S.R.L.');
insert into CLIENT values(seq_client.nextval, 'fizic', null);
insert into CLIENT values(seq_client.nextval, 'fizic', null);
insert into CLIENT values(seq_client.nextval, 'fizic', 'Evenimente cu
Familia');
insert into CLIENT values(seq_client.nextval, 'online', 'Agentia de imobiliare
BuyToday');
insert into CLIENT values(seq_client.nextval, 'online', null);
insert into CLIENT values(seq_client.nextval, 'online', 'Alpha Bank
Headquarters');
insert into CLIENT values(seq_client.nextval, 'fizic', null);

insert into CLIENT_ONLINE values(10000000, 'alexandrescu_a@gmail.com', null,
'Andra');
```

```

insert into CLIENT_ONLINE values(10000020,
'andreea.badea@retailexpress.outlook.com', 'Badea', 'Andreea');
insert into CLIENT_ONLINE values(10000030, 'alina.popescu1997@gmail.com',
'Popescu', 'Alina');
insert into CLIENT_ONLINE values(10000070,
'dan_cristescu@manager.buytoday.ro', 'Cristescu', 'Dan');
insert into CLIENT_ONLINE values(10000080, 'alex_dragos@gmail.com', null,
'Alexandru');
insert into CLIENT_ONLINE values(10000090,
'maria_pestritu@alphabank.outlook.ro', 'Pestritu', 'Maria');

insert into CLIENT_FIZIC values(10000010, '+40772937066');
insert into CLIENT_FIZIC values(10000040, '+40791816617');
insert into CLIENT_FIZIC values(10000050, '+40716423256');
insert into CLIENT_FIZIC values(10000060, '+40756246891');
insert into CLIENT_FIZIC values(10000100, '+40752789141');

-- magazine
insert into ADRESA values(032266, 'Bucuresti', 'Romania', 'Bulevardul Theodor Pallady, nr. 57, Sector 3');
insert into ADRESA values(013696, 'Bucuresti', 'Romania', 'Sos. Bucuresti Ploiesti, nr. 42A, Sector 1');
insert into ADRESA values(307160, 'Timisoara', 'Romania', 'Bulevardul Petre Tutea, nr. 2A');
insert into ADRESA values(400436, 'Cluj-Napoca', 'Romania', 'Str. Alexandru Vaida Voevod 53B');
insert into ADRESA values(500238, 'Brasov', 'Romania', 'Str. Crisului, nr. 18');
insert into ADRESA values(707252, 'Iasi', 'Romania', 'Bulevardul Calea Chisinaului, nr. 29');
insert into ADRESA values(410224, 'Oradea', 'Romania', 'Calea Aradului 87A');
insert into ADRESA values(550088, 'Sibiu', 'Romania', 'Calea Sului Mari 43');
insert into ADRESA values(900147, 'Constanta', 'Romania', 'Str. Cumpenei, nr. 2');

-- clienti
insert into ADRESA values(040394, 'Bucuresti', 'Romania', 'Str. Visana, nr. 3, bloc 44, sc. A, ap. 12');
insert into ADRESA values(700259, 'Iasi', 'Romania', 'Str. Vasile Stroescu, nr. 28, bloc Y4');
insert into ADRESA values(106100, 'Sinaia', 'Romania', 'Str. Paraul Dorului');
insert into ADRESA values(010566, 'Bucuresti', 'Romania', 'Calea Dorobantilor, nr. 237 B, Sector 1');

insert into MAGAZIN values('A42D', '+40789947067', 0, 032266);
insert into MAGAZIN values('090P', '+40769937527', 0, 307160);
insert into MAGAZIN values('45E', '+40781550674', 0, 013696);
insert into MAGAZIN values('234Y', '+40752356779', 1, 550088);
insert into MAGAZIN values('RT56', '+40742622774', 1, 400436);

```

```
insert into MAGAZIN values('7TY6', '+40782378375', 1, 707252);
insert into MAGAZIN values('10UP', '+40792378237', 1, 900147);
insert into MAGAZIN values('56GH', '+40721729803', 1, 500238);
insert into MAGAZIN values('4DF', '+40762751657', 1, 410224);

insert into AGENT_VANZARI values(seq_agent.nextval, 'Adela', 'Angelescu', '19-MARCH-22', '+40785267887', 'A42D');
insert into AGENT_VANZARI values(seq_agent.nextval, 'Madalin', 'Stroe', '09-FEBRUARY-21', '+40790380926', 'A42D');
insert into AGENT_VANZARI values(seq_agent.nextval, 'Andrei', 'Comanescu', '18-MAY-21', '+40799080928', '45E');
insert into AGENT_VANZARI values(seq_agent.nextval, 'Mircea', 'Istrate', '15-APRIL-20', '+40728767134', '090P');
insert into AGENT_VANZARI values(seq_agent.nextval, 'Daria', 'Marculescu', '17-FEBRUARY-21', '+40754678239', '45E');
insert into AGENT_VANZARI values(seq_agent.nextval, 'Daniela', 'Ignat', '10-OCTOBER-23', '+40754156824', null);
insert into AGENT_VANZARI values(seq_agent.nextval, 'Mihai', 'Radulescu', '07-JANUARY-24', '+40778567230', null);

insert into COMANDA values(seq_comanda.nextval, 793.82, '24-MAY-23', 10020, 10000040, 040394);
insert into COMANDA values(seq_comanda.nextval, 1574.51, '05-DECEMBER-21', 10000, 10000100, null);
insert into COMANDA values(seq_comanda.nextval, 3460.05, '18-APRIL-24', 10000, 10000080, 707252);
insert into COMANDA values(seq_comanda.nextval, 2388.15, '11-FEBRUARY-24', null, 10000080, 700259);
insert into COMANDA values(seq_comanda.nextval, 679.3, '28-MAY-24', 10010, 10000050, null);
insert into COMANDA values(seq_comanda.nextval, 349.75, '29-MAY-24', 10025, 10000050, null);
insert into COMANDA values(seq_comanda.nextval, 5667.75, '14-MAY-24', null, 10000030, 106100);
insert into COMANDA values(seq_comanda.nextval, 3853.05, '28-MAY-24', 10025, 10000000, 400436);
insert into COMANDA values(seq_comanda.nextval, 919.7, '30-AUGUST-21', 10030, 10000020, 900147);
insert into COMANDA values(seq_comanda.nextval, 9751.92, '19-NOVEMBER-23', 10005, 10000010, null);
insert into COMANDA values(seq_comanda.nextval, 1439.03, '21-JUNE-23', 10015, 10000060, null);
insert into COMANDA values(seq_comanda.nextval, 8144.95, '28-MAY-24', 10005, 10000070, 550088);
insert into COMANDA values(seq_comanda.nextval, 1199.98, '09-FEBRUARY-24', null, 10000090, 010566);
insert into COMANDA values(seq_comanda.nextval, 4192.55, '28-MAY-24', 10000, 10000090, 010566);
```

```
insert into COMANDA values(seq_comanda.nextval, 5349.75, '13-MAY-2018', 10030,  
10000040, null);  
insert into COMANDA values(seq_comanda.nextval, 249.99, '13-MAY-2018', null,  
10000100, null);  
insert into COMANDA values(seq_comanda.nextval, 0, '12-JUN-19', null,  
10000030, null);  
insert into COMANDA values(seq_comanda.nextval, 0, '09-JUN-20', null,  
10000080, null);  
insert into COMANDA values(seq_comanda.nextval, 0, '19-OCT-21', null,  
10000070, null);  
insert into COMANDA values(seq_comanda.nextval, 0, '12-OCT-20', null,  
10000080, null);  
insert into COMANDA values(seq_comanda.nextval, 0, '23-OCT-20', null,  
10000040, null);  
insert into COMANDA values(seq_comanda.nextval, 0, '27-OCT-22', null,  
10000050, null);  
insert into COMANDA values(seq_comanda.nextval, 0, '29-OCT-24', null,  
10000090, null);  
  
insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'in procesare',  
1000000000);  
insert into TRANZACTIE values(seq_tranzactie.nextval, 'cash', 'aprobata',  
1000000002);  
insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'aprobata',  
1000000004);  
insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'respinsa',  
1000000006);  
insert into TRANZACTIE values(seq_tranzactie.nextval, 'cash', 'aprobata',  
1000000008);  
insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'in procesare',  
1000000010);  
insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'aprobata',  
1000000012);  
insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'in procesare',  
1000000014);  
insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'aprobata',  
1000000016);  
insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'aprobata',  
1000000018);  
insert into TRANZACTIE values(seq_tranzactie.nextval, 'cash', 'aprobata',  
1000000020);  
insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'respinsa',  
1000000022);  
insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'aprobata',  
1000000024);  
insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'aprobata',  
1000000026);
```

```

insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'aprobata',
1000000028);
insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'aprobata',
1000000030);
insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'aprobata',
1000000032);
insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'aprobata',
1000000034);
insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'respinsa',
1000000036);
insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'aprobata',
1000000038);
insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'aprobata',
1000000040);
insert into TRANZACTIE values(seq_tranzactie.nextval, 'cash', 'aprobata',
1000000042);
insert into TRANZACTIE values(seq_tranzactie.nextval, 'card', 'aprobata',
1000000044);

insert into CATEGORIE values('ASDRT', 'Accesorii');
insert into CATEGORIE values('PASDU', 'Paturi');
insert into CATEGORIE values('GHJeR', 'Rafturi, dulapuri si unitati de
depozitare');
insert into CATEGORIE values('DFGhT', 'Unitati de dulapuri pentru bucatarie');
insert into CATEGORIE values('SDsRK', 'Scaune, mese si birouri');
insert into CATEGORIE values('madYU', 'Mobilier de exterior');
insert into CATEGORIE values('CghUI', 'Canapele si fotolii');
insert into CATEGORIE values('kkRTy', 'Jucarii si jocuri');
insert into CATEGORIE values('eedaT', 'Gradina');

insert into PIESA_MOBILIER values(seq_produs.nextval, 'MALM', 1999.89, 'Pat cu
spatiu de depozitare', 210, 175, 40, '4 ani, prelungire 1 an',
'https://www.ikea.com/ro/ro/p/malm-pat-cu-depozitare-alb-20404806/', 'Eva
Lilja Lowenhielm', 'PASDU');
insert into PIESA_MOBILIER values(seq_produs.nextval, 'IDANAS', 2499.15, 'Pat
tapitat cu depozitare, Gunnared gri inchis, 140x200 cm', 224, 150, 40, '3 ani,
fara prelungire', 'https://www.ikea.com/ro/ro/p/idanaes-pat-tapitat-cu-
depozitare-gunnared-gri-inchis-40458964/', 'Francis Cayouette', 'PASDU');
insert into PIESA_MOBILIER values(seq_produs.nextval, 'ENHET', 50.50, 'Polita
rotativa, antracit, 40x21 cm', 21, 21, 40, null,
'https://www.ikea.com/ro/ro/p/enhet-polita-rotativa-antracit-20465734/', 'IKEA
of Sweden/E Lilja Lowenhielm', 'ASDRT');
insert into PIESA_MOBILIER values(seq_produs.nextval, 'KLIPSK', 39.90, 'Tava
mic-dejun, gri', 56, 36, 26, null, 'https://www.ikea.com/ro/ro/p/klipsk-tava-
mic-dejun-gri-10327700/', 'Marcus Arvonen', 'ASDRT');
insert into PIESA_MOBILIER values(seq_produs.nextval, 'EKET', 320.30, 'Corp cu
2 usi si 1 polita, gri inchis, 70x35x70 cm', 70, 35, 70, '2 ani, fara

```

```
prelungire', 'https://www.ikea.com/ro/ro/p/eket-corp-cu-2-usi-si-1-polita-gri-inchis-20344921/', 'Jon Karlsson', 'GHJeR');
```

```
insert into PIESA_MOBILIER values(seq_produs.nextval, 'BILLY', 249.45, 'Biblioteca, alb, 80x28x202 cm', 80, 28, 202, '1 an, prelungire 1 an', 'https://www.ikea.com/ro/ro/p/billy-biblioteca-alb-00263850/', 'Gillis Lundgren', 'GHJeR');
```

```
insert into PIESA_MOBILIER values(seq_produs.nextval, 'KALLAX', 749.20, 'Etajera cu 4 organizatoare, negru-maro, 147x147 cm', 147, 39, 147, '2 ani, fara prelungire', 'https://www.ikea.com/ro/ro/p/kallax-etajera-cu-4-organizatoare-negru-maro-s09017483/', 'Tord Bjorklund', 'GHJeR');
```

```
insert into PIESA_MOBILIER values(seq_produs.nextval, 'FINNBY', 229.90, 'Biblioteca, negru, 60x180 cm', 60, 24, 180, '2 ani, fara prelungire', 'https://www.ikea.com/ro/ro/p/finnby-biblioteca-negru-10261129/', 'IKEA of Sweden', 'GHJeR');
```

```
insert into PIESA_MOBILIER values(seq_produs.nextval, 'MARKUS', 599.99, 'Scaun rotativ, Vissle gri inchis', 62, 60, 140, '10 ani', 'https://www.ikea.com/ro/ro/p/markus-scaun-rotativ-vissle-gri-inchis-70261150/', 'Henrik Preutz', 'SDsRK');
```

```
insert into PIESA_MOBILIER values(seq_produs.nextval, 'BERGMUND', 349.75, 'Scaun, negru/Gunnared gri mediu', 52, 59, 96, null, 'https://www.ikea.com/ro/ro/p/bergmund-scaun-negru-gunnared-gri-mediu-s69384307/', 'IKEA of Sweden/K Hagberg/M Hagberg', 'SDsRK');
```

```
insert into PIESA_MOBILIER values(seq_produs.nextval, 'SKOGSTA/INGOLF', 3433.50, 'Masa+6scaune, acacia/negru, 235x100 cm', 235, 100, 74, '5 ani, prelungire 1 an', 'https://www.ikea.com/ro/ro/p/skogsta-ingolf-masa-6scaune-acacia-negru-s09482693/', null, 'SDsRK');
```

```
insert into PIESA_MOBILIER values(seq_produs.nextval, 'TARSELE', 2299.20, 'Masa extensibila, furnir stejar/negru, 150/200x80 cm', 200, 80, 77, '3 ani, fara prelungire', 'https://www.ikea.com/ro/ro/p/tarsele-masa-extensibila-furnir-stejar-negru-70581359/', null, 'SDsRK');
```

```
insert into PIESA_MOBILIER values(seq_produs.nextval, 'MITTCIRKEL/ALEX', 739.99, 'Birou, aspect pin intens/alb, 140x60 cm', 140, 60, 73, '2 ani, prelungire 1 an', 'https://www.ikea.com/ro/ro/p/mittcirkel-alex-birou-aspect-pin-intens-alb-s09521722/', null, 'SDsRK');
```

```
insert into PIESA_MOBILIER values(seq_produs.nextval, 'BRIMNES', 799.15, 'Dulap cu 3 usi, negru, 117x190 cm', 117, 50, 190, '2 ani, fara prelungire', 'https://www.ikea.com/ro/ro/p/brimnes-dulap-cu-3-usi-negru-60407577/', 'K Hagberg/M Hagberg', 'GHJeR');
```

```
insert into PIESA_MOBILIER values(seq_produs.nextval, 'KNOXHULT', 1889.25, 'Bucatarie, alb, 220x61x220 cm', 220, 61, 220, '7 ani garantie, fara prelungire', 'https://www.ikea.com/ro/ro/p/knoxhult-bucatarie-alb-s49180467/', 'IKEA of Sweden/Mikael Warnhammar', 'DFGhT');
```

```
insert into PIESA_MOBILIER values(seq_produs.nextval, 'METOD/MAXIMERA', 419.55, 'Corp baza plita+cupitor+sertar, alb/Bodarp gri-verde, 60x60 cm', 60, 61, 88, null, 'https://www.ikea.com/ro/ro/p/metod-maximera-corp-baza-plita-cupitor-sertar-alb-bodarp-gri-verde-s19306816/', null, 'DFGhT');
```

```
insert into PIESA_MOBILIER values(seq_produs.nextval, 'ASPINGE', 2388.15, 'Chicineta, negru/frasin, 120x60x202 cm', 120, 60, 202, '10 ani, fara
```

```

prelungire', 'https://www.ikea.com/ro/ro/p/aespinge-chicineta-negru-frasin-s99478168/', null, 'DFGhT');
insert into PIESA_MOBILIER values(seq_produs.nextval, 'LANDSKRONA', 2499.15,
'Canapea 3 locuri, Gunnared verde deschis/lemn', 204, 89, 78, '3 ani,
prelungire 2 ani', 'https://www.ikea.com/ro/ro/p/landskrona-canapea-3-locuri-gunnared-verde-deschis-lemn-s39270326/', 'IKEA of Sweden/Tord Bj?rklund',
'CghUI');
insert into PIESA_MOBILIER values(seq_produs.nextval, 'LANGARYD', 6499.99,
'Canapea 3 locuri+sezlong, dreapta, Lejde gri/negru/lemn', 280, 135, 70, '5 ani,
prelungire 2 ani', 'https://www.ikea.com/ro/ro/p/langaryd-canapea-3locuri-sezlong-dreapta-lejde-gri-negru-lemn-s19418734/', null, 'CghUI');
insert into PIESA_MOBILIER values(seq_produs.nextval, 'KIVIK', 1449.55,
'Canapea 2 locuri, Tibbleby bej/gri', 190, 95, 83, '4 ani, fara prelungire',
'https://www.ikea.com/ro/ro/p/kivik-canapea-2-locuri-tibbleby-bej-gri-s09440599/', null, 'CghUI');
insert into PIESA_MOBILIER values(seq_produs.nextval, 'SODERHAMN', 1700.10,
'Sezlong, Viarp bej/maro', 93, 151, 83, '1 an, fara prelungire',
'https://www.ikea.com/ro/ro/p/soederhamn-sezlong-viarp-bej-maro-s89305620/',
'Ola Wihlborg', 'CghUI');
insert into PIESA_MOBILIER values(seq_produs.nextval, 'ASKHOLMEN', 149.35,
'Masa pentru perete, exterior, Pliant maro inchis, 70x44 cm', 70, 44, 71,
null, 'https://www.ikea.com/ro/ro/p/askholmen-masa-pentru-perete-exterior-pliant-maro-inchis-70557496/', 'Jon Karlsson', 'madYU');
insert into PIESA_MOBILIER values(seq_produs.nextval, 'NAMMARO', 449.99,
'Banca cu spatar, exterior, maro deschis vopsit', 62, 78, 50, null,
'https://www.ikea.com/ro/ro/p/naemmaroe-banca-cu-spatar-exterior-maro-deschis-vopsit-30510302/', 'Nike Karlsson', 'madYU');
insert into PIESA_MOBILIER values(seq_produs.nextval, 'DJUNGELSKOG', 129.90,
'Jucarie de plus, urs brun', 100, 50, 50, null,
'https://www.ikea.com/ro/ro/p/djungelskog-jucarie-de-plus-urs-brun-00402813/',
'Annie Hulden', 'kkRTy');
insert into PIESA_MOBILIER values(seq_produs.nextval, 'SATSUMAS', 249.99,
'Suport plante, bambus/alb, 70 cm', 84, 28, 70, null,
'https://www.ikea.com/ro/ro/p/satsumas-suport-plante-bambus-alb-90258156/',
'Carl Hagerling', 'eedaT');

insert into ADAUGA_COMANDA values(400023, 1000000008, 1, '28-MAY-24 10:20:00');
insert into ADAUGA_COMANDA values(400002, 1000000008, 1, '28-MAY-24 10:20:03');
insert into ADAUGA_COMANDA values(400005, 1000000008, 2, '28-MAY-24 10:20:04');
insert into ADAUGA_COMANDA values(400009, 1000000010, 1, '28-MAY-24 10:20:06');
insert into ADAUGA_COMANDA values(400010, 1000000014, 1, '28-MAY-24 10:20:08');
insert into ADAUGA_COMANDA values(400015, 1000000014, 1, '28-MAY-24 10:20:09');

```

```
insert into ADAUGA_COMANDA values(400006, 1000000022, 1, '28-MAY-24  
10:20:11');  
insert into ADAUGA_COMANDA values(400017, 1000000022, 2, '28-MAY-24  
10:20:12');  
insert into ADAUGA_COMANDA values(400013, 1000000022, 3, '28-MAY-24  
10:20:14');  
insert into ADAUGA_COMANDA values(400012, 1000000026, 1, '28-MAY-24  
10:20:16');  
insert into ADAUGA_COMANDA values(400009, 1000000026, 4, '28-MAY-24  
10:20:17');  
insert into ADAUGA_COMANDA values(400008, 1000000026, 2, '28-MAY-24  
10:20:20');  
insert into ADAUGA_COMANDA values(400007, 1000000026, 5, '28-MAY-24  
10:20:21');  
insert into ADAUGA_COMANDA values(400009, 1000000000, 1, '24-MAY-23  
10:37:10');  
insert into ADAUGA_COMANDA values(400012, 1000000000, 1, '24-MAY-23  
10:40:30');  
insert into ADAUGA_COMANDA values(400003, 1000000002, 2, '05-DEC-21  
21:15:20');  
insert into ADAUGA_COMANDA values(400011, 1000000002, 1, '05-DEC-21  
21:51:30');  
insert into ADAUGA_COMANDA values(400001, 1000000004, 1, '18-APR-24  
19:22:35');  
insert into ADAUGA_COMANDA values(400004, 1000000004, 3, '18-APR-24  
20:00:10');  
insert into ADAUGA_COMANDA values(400016, 1000000006, 1, '11-FEB-24  
12:06:10');  
insert into ADAUGA_COMANDA values(400014, 1000000012, 3, '14-MAY-24  
17:09:45');  
insert into ADAUGA_COMANDA values(400007, 1000000016, 5, '30-AUG-21  
01:45:35');  
insert into ADAUGA_COMANDA values(400018, 1000000018, 3, '19-NOV-23  
18:07:10');  
insert into ADAUGA_COMANDA values(400023, 1000000020, 3, '21-JUN-23  
16:05:11');  
insert into ADAUGA_COMANDA values(400022, 1000000020, 2, '21-JUN-23  
16:35:50');  
insert into ADAUGA_COMANDA values(400021, 1000000020, 1, '21-JUN-23  
17:40:30');  
insert into ADAUGA_COMANDA values(400008, 1000000024, 2, '09-FEB-24  
09:23:30');  
insert into ADAUGA_COMANDA values(400020, 1000000028, 3, '13-MAY-18  
10:09:45');  
insert into ADAUGA_COMANDA values(400005, 1000000028, 1, '13-MAY-18  
11:09:45');  
insert into ADAUGA_COMANDA values(400024, 1000000030, 1, '13-MAY-18  
07:19:46');
```

```

insert into ADAUGA_COMANDA values(400015, 1000000032, 3, '12-JUN-19
10:20:09');
insert into ADAUGA_COMANDA values(400006, 1000000032, 2, '12-JUN-19
12:20:11');
insert into ADAUGA_COMANDA values(400009, 1000000034, 4, '09-JUN-20
09:40:09');
insert into ADAUGA_COMANDA values(400021, 1000000034, 1, '09-JUN-20
11:45:12');
insert into ADAUGA_COMANDA values(400014, 1000000036, 1, '19-OCT-21
08:20:09');
insert into ADAUGA_COMANDA values(400022, 1000000036, 2, '19-OCT-21
12:20:11');
insert into ADAUGA_COMANDA values(400001, 1000000038, 1, '12-OCT-20
07:20:09');
insert into ADAUGA_COMANDA values(400023, 1000000038, 1, '12-OCT-20
09:20:11');
insert into ADAUGA_COMANDA values(400014, 1000000040, 1, '23-OCT-20
10:50:12');
insert into ADAUGA_COMANDA values(400003, 1000000042, 3, '27-OCT-22
12:27:11');
insert into ADAUGA_COMANDA values(400021, 1000000044, 1, '29-OCT-24
10:10:50');

insert into OFERTA values(400002, '28-MAY-24', '01-JUNE-24', 24.99);
insert into OFERTA values(400010, '28-MAY-24', '02-JUNE-24', 12.99);
insert into OFERTA values(400007, '28-MAY-24', '07-JUNE-24', 29.99);
insert into OFERTA values(400012, '23-MAY-23', '29-MAY-24', 39.99);
insert into OFERTA values(400011, '01-DECEMBER-21', '10-DECEMBER-21', 34.99);
insert into OFERTA values(400018, '19-NOVEMBER-23', '19-NOVEMBER-23', 49.99);
insert into OFERTA values(400007, '30-AUGUST-21', '10-SEPTEMBER-21', 19.99);
insert into OFERTA values(400003, '04-DECEMBER-21', '10-DECEMBER-21', 20);

insert into FURNIZOR values('ADS', 'Mircea Oprea', '+40794267037');
insert into FURNIZOR values('AER3', 'Mihnea Vasilescu', '+40712347037');
insert into FURNIZOR values('67RT', 'Adriana Georgescu', '+40794234567');
insert into FURNIZOR values('0003', 'Petru Coman', '+40791234567');
insert into FURNIZOR values('1S', 'Daniel Popara', '+40778967037');

insert into MATERIE_PRIMA values(seq_material.nextval, 'MDF', 'metru patrat',
20.37, '0003');
insert into MATERIE_PRIMA values(seq_material.nextval, 'MDF', 'centimetru
patrat', 5.47, '1S');
insert into MATERIE_PRIMA values(seq_material.nextval, 'Plastic', 'metru cub',
2.37, 'AER3');
insert into MATERIE_PRIMA values(seq_material.nextval, 'PAL', 'metru patrat',
7.89, '1S');
insert into MATERIE_PRIMA values(seq_material.nextval, 'Vopsea', 'decimetru
cub', 3.44, '1S');

```

```
insert into MATERIE_PRIMA values(seq_material.nextval, 'Lemn masiv', 'metru patrat', 50.99, '67RT');
insert into MATERIE_PRIMA values(seq_material.nextval, 'Spuma poliuretanica', 'centimetru cub', 12.34, '1S');
insert into MATERIE_PRIMA values(seq_material.nextval, 'Otel galvanizat', 'kilogram', 6.78, '1S');
insert into MATERIE_PRIMA values(seq_material.nextval, 'Hartie laminata', 'kilogram', 3.21, 'ADS');
insert into MATERIE_PRIMA values(seq_material.nextval, 'Lac acrilic', 'gram', 1.32, '0003');
insert into MATERIE_PRIMA values(seq_material.nextval, 'Otel galvanizat', 'kilogram', 5.47, 'ADS');
insert into MATERIE_PRIMA values(seq_material.nextval, 'Poliester', 'kilogram', 10.32, '67RT');
insert into MATERIE_PRIMA values(seq_material.nextval, 'Bumbac', 'centimetru cub', 5.47, 'ADS');

insert into PRODUSA_DIN values(400000, 5000000003, 6);
insert into PRODUSA_DIN values(400000, 5000000006, 3);
insert into PRODUSA_DIN values(400000, 5000000009, 10);
insert into PRODUSA_DIN values(400000, 5000000012, 1);
insert into PRODUSA_DIN values(400001, 5000000003, 6);
insert into PRODUSA_DIN values(400001, 5000000009, 5);
insert into PRODUSA_DIN values(400001, 5000000015, 3);
insert into PRODUSA_DIN values(400001, 5000000018, 1);
insert into PRODUSA_DIN values(400001, 5000000030, 4);
insert into PRODUSA_DIN values(400002, 5000000021, 2);
insert into PRODUSA_DIN values(400008, 5000000033, 3);
insert into PRODUSA_DIN values(400008, 5000000018, 5);
insert into PRODUSA_DIN values(400008, 5000000006, 2);
insert into PRODUSA_DIN values(400003, 5000000006, 1);
insert into PRODUSA_DIN values(400004, 5000000000, 4);
insert into PRODUSA_DIN values(400004, 5000000024, 1);
insert into PRODUSA_DIN values(400004, 5000000009, 3);
insert into PRODUSA_DIN values(400004, 5000000006, 6);
insert into PRODUSA_DIN values(400011, 5000000027, 2);
insert into PRODUSA_DIN values(400017, 5000000030, 3);
insert into PRODUSA_DIN values(400017, 5000000033, 6);
insert into PRODUSA_DIN values(400017, 5000000036, 2);--
insert into PRODUSA_DIN values(400007, 5000000009, 5);
insert into PRODUSA_DIN values(400021, 5000000003, 14);
insert into PRODUSA_DIN values(400024, 5000000021, 2);

insert into STOC values('AGFJK', '01-DECEMBER-21');
insert into STOC values('DFGTY', '14-MAY-24');
insert into STOC values('BSJUY', '18-OCTOBER-22');
insert into STOC values('RLJFG', '12-AUGUST-19');
insert into STOC values('OUNDS', '15-DECEMBER-24');
```

```

insert into STOC values('AGFJK', '13-APRIL-22');
insert into STOC values('RLJFG', '02-MARCH-19');
insert into STOC values('DFGTY', '29-MAY-24');
insert into STOC values('OUNDS', '01-JUNE-24');
insert into STOC values('BSJUY', '03-JUNE-24');
insert into STOC values('DFGTY', '24-MAY-24');
insert into STOC values('AGFJK', '06-DECEMBER-21');
insert into STOC values('DFGTY', '07-SEPTEMBER-21');
insert into STOC values('OUNDS', '09-DECEMBER-21');

insert into APROVIZIONEAZA values('56GH', 400009, 'AGFJK', '01-DEC-21', 5);
insert into APROVIZIONEAZA values('10UP', 400015, 'BSJUY', '18-OCT-22', 21);
insert into APROVIZIONEAZA values('10UP', 400005, 'BSJUY', '18-OCT-22', 137);
insert into APROVIZIONEAZA values('090P', 400009, 'RLJFG', '12-AUG-19', 99);
insert into APROVIZIONEAZA values('234Y', 400021, 'RLJFG', '12-AUG-19', 23);
insert into APROVIZIONEAZA values('45E', 400010, 'RLJFG', '12-AUG-19', 4);
insert into APROVIZIONEAZA values('10UP', 400003, 'OUNDS', '15-DEC-24', 78);
insert into APROVIZIONEAZA values('234Y', 400017, 'OUNDS', '15-DEC-24', 35);
insert into APROVIZIONEAZA values('090P', 400019, 'AGFJK', '13-APR-22', 35);
insert into APROVIZIONEAZA values('4DF', 400023, 'AGFJK', '13-APR-22', 27);
insert into APROVIZIONEAZA values('7TY6', 400017, 'RLJFG', '02-MAR-19', 14);
insert into APROVIZIONEAZA values('56GH', 400000, 'BSJUY', '18-OCT-22', 67);
insert into APROVIZIONEAZA values('56GH', 400002, 'DFGTY', '29-MAY-24', 3);
insert into APROVIZIONEAZA values('090P', 400010, 'OUNDS', '1-JUN-24', 10);
insert into APROVIZIONEAZA values('234Y', 400007, 'BSJUY', '3-JUN-24', 25);
insert into APROVIZIONEAZA values('45E', 400012, 'DFGTY', '24-MAY-24', 1);
insert into APROVIZIONEAZA values('10UP', 400011, 'AGFJK', '6-DEC-21', 34);
insert into APROVIZIONEAZA values('234Y', 400007, 'DFGTY', '7-SEP-21', 35);
insert into APROVIZIONEAZA values('090P', 400003, 'OUNDS', '9-DEC-21', 68);

```

The screenshot shows the Microsoft SQL Server Management Studio (SSMS) interface. On the left, the Object Explorer pane displays a tree view of the database structure under 'Tables (Filtered)'. The tables listed are ADAUGA_COMANDA, ADRESA, AGENT_VANZARI, APROVIZIONEAZA, CATEGORIE, CLIENT, CLIENT_FIZIC, and CLIENT_ONLINE. The central area contains a large block of SQL code for inserting data into the APROVIZIONEAZA table. Below the code, the 'Script Output' pane shows five lines of text indicating successful insertions: '1 row inserted.' repeated five times.

```

Insert into APROVIZIONEAZA

```

Script Output X

Task completed in 1 row inserted.

Tabela CLIENT

	ID_CLIENT	TIP_CLIENT	NUME_FIRMA
1	10000000	online	(null)
2	10000010	fizic	DEluxe Store
3	10000020	online	Retail Express S.R.L.
4	10000030	online	Alina CraftShop S.R.L.
5	10000040	fizic	(null)
6	10000050	fizic	(null)
7	10000060	fizic	Evenimente cu Familia
8	10000070	online	Agentia de imobiliare BuyToday
9	10000080	online	(null)
10	10000090	online	Alpha Bank Headquarters
11	10000100	fizic	(null)

Tabela CLIENT_ONLINE

	ID_CLIENT	EMAIL	NUME	PRENUME
1	10000000	alexandrescu_a@gmail.com	(null)	Andra
2	10000020	andreea.badea @_retailexpress.outlook.com	Badea	Andreea
3	10000030	alinapopescu997@gmail.com	Popescu	Alina
4	10000070	dan_cristescu@manager.buytoday.ro	Cristescu	Dan
5	10000080	alex_dragos@gmail.com	(null)	Alexandru
6	10000090	maria_pestraru@alphabank.outlook.ro	Pestraru	Maria

Tabela CLIENT_FIZIC

	ID_CLIENT	TELEFON
1	10000010	+40772937066
2	10000040	+40791816617
3	10000050	+40716423256
4	10000060	+40756246891
5	10000100	+40752789141

Tabela ADRESA

	COD_POSTAL	ORAS	TARA	STRADA
1	32266	Bucuresti	Romania	Bulevardul Theodor Pallady, nr. 57, Sector 3
2	13696	Bucuresti	Romania	Sos. Bucuresti Ploiesti, nr. 42A, Sector 1
3	307160	Timisoara	Romania	Bulevardul Petre Tutea, nr. 2A
4	400436	Cluj-Napoca	Romania	Str. Alexandru Vaida Voievod 53B
5	500238	Brasov	Romania	Str. Crisului, nr. 18
6	707252	Iasi	Romania	Bulevardul Calea Chisinaului, nr. 29
7	410224	Oradea	Romania	Calea Aradului 87A
8	550088	Sibiu	Romania	Calea Sului Mari 43
9	900147	Constanta	Romania	Str. Cumpenei, nr. 2
10	40394	Bucuresti	Romania	Str. Visana, nr. 3, bloc 44, sc. A, ap. 12
11	700259	Iasi	Romania	Str. Vasile Stroescu, nr. 28, bloc Y4
12	106100	Sinaia	Romania	Str. Paraul Dorului
13	10566	Bucuresti	Romania	Calea Dorobantilor, nr. 237 B, Sector 1

Tabela MAGAZIN

	ID_MAGAZIN	TELEFON	CENTRU RIDICARE	COD_POSTAL
1	A42D	+40789947067	0	32266
2	090P	+40769937527	0	307160
3	45E	+40781550674	0	13696
4	234Y	+40752356779	1	550088
5	RT56	+40742622774	1	400436
6	7TY6	+40782378375	1	707252
7	10UP	+40792378237	1	900147
8	56GH	+40721729803	1	500238
9	4DF	+40762751657	1	410224

Tabela AGENT_VANZARI

	ID_ANGAJAT	NUME	PRENUME	DATA_ANGAJARE	TELEFON	ID_MAGAZIN
1	10000	Adela	Angelescu	19-MAR-22	+40785267887	A42D
2	10005	Madalin	Stroe	09-FEB-21	+40790380926	A42D
3	10010	Andrei	Comanescu	18-MAY-21	+40799080928	45E
4	10015	Mircea	Istrate	15-APR-20	+40728767134	090P
5	10020	Daria	Marculescu	17-FEB-21	+40754678239	45E
6	10025	Daniela	Ignat	10-OCT-23	+40754156824	(null)
7	10030	Mihai	Radulescu	07-JAN-24	+40778567230	(null)

Tabela COMANDA

	ID_COMANDA	PRET	DATA_ACHIZITIE	ID_ANGAJAT	ID_CLIENT	COD_POSTAL
1	1000000000	793.82	24-MAY-23	10020	10000040	40394
2	1000000002	1574.51	05-DEC-21	10000	10000100	(null)
3	1000000004	3460.05	18-APR-24	10000	10000080	707252
4	1000000006	2388.15	11-FEB-24	(null)	10000080	700259
5	1000000008	679.3	28-MAY-24	10010	10000050	(null)
6	1000000010	349.75	29-MAY-24	10025	10000050	(null)
7	1000000012	5667.75	14-MAY-24	(null)	10000030	106100
8	1000000014	3853.05	28-MAY-24	10025	10000000	400436
9	1000000016	919.7	30-AUG-21	10030	10000020	900147
10	1000000018	9751.92	19-NOV-23	10005	10000010	(null)
11	1000000020	1439.03	21-JUN-23	10015	10000060	(null)
12	1000000022	8144.95	28-MAY-24	10005	10000070	550088
13	1000000024	1199.98	09-FEB-24	(null)	10000090	10566
14	1000000026	4192.55	28-MAY-24	10000	10000090	10566
15	1000000028	5349.75	13-MAY-18	10030	10000040	(null)
16	1000000030	249.99	13-MAY-18	(null)	10000100	(null)
17	1000000032	0	12-JUN-19	(null)	10000030	(null)
18	1000000034	0	09-JUN-20	(null)	10000080	(null)
19	1000000036	0	19-OCT-21	(null)	10000070	(null)
20	1000000038	0	12-OCT-20	(null)	10000080	(null)
21	1000000040	0	23-OCT-20	(null)	10000040	(null)
22	1000000042	0	27-OCT-22	(null)	10000050	(null)
23	1000000044	0	29-OCT-24	(null)	10000090	(null)

Tabela TRANZACTIE

	ID_TRANZACTIE	MODALITATE_PLATA	STATUS_PLATA	ID_COMANDA
1	1000000001	card	in procesare	1000000000
2	1000000003	cash	aprobata	1000000002
3	1000000005	card	aprobata	1000000004
4	1000000007	card	respinsa	1000000006
5	1000000009	cash	aprobata	1000000008
6	1000000011	card	in procesare	1000000010
7	1000000013	card	aprobata	1000000012
8	1000000015	card	in procesare	1000000014
9	1000000017	card	aprobata	1000000016
10	1000000019	card	aprobata	1000000018
11	1000000021	cash	aprobata	1000000020
12	1000000023	card	respinsa	1000000022
13	1000000025	card	aprobata	1000000024
14	1000000027	card	aprobata	1000000026
15	1000000029	card	aprobata	1000000028
16	1000000031	card	aprobata	1000000030
17	1000000033	card	aprobata	1000000032
18	1000000035	card	aprobata	1000000034
19	1000000037	card	respinsa	1000000036
20	1000000039	card	aprobata	1000000038
21	1000000041	card	aprobata	1000000040
22	1000000043	cash	aprobata	1000000042
23	1000000045	card	aprobata	1000000044

Tabela CATEGORIE

	ID_CATEGORIE	NUME
1	ASDRT	Accesorii
2	PASDU	Paturi
3	GHJeR	Rafturi, dulapuri si unitati de depozitare
4	DFGhT	Unitati de dulapuri pentru bucatarie
5	SDsRK	Scaune, mese si birouri
6	madYU	Mobilier de exterior
7	CghUI	Canapele si fotolii
8	kkRTy	Jucarii si jocuri
9	eedaT	Gradina

Tabela PIEZA_MOBILIER

	ID_PRODUS	NUME	PRET	DESCRIERE	LUNGIME	LATIME	INALTIME
1	400000	MALM	1999.89	Pat cu spatiu de depozitare	210	175	40.4
2	400001	IDANAS	2499.15	Pat tapitat cu depozitare, Gunnared gri inchis, 140x200 cm	224	150	40.3
3	400002	ENHET	50.5	Polita rotativa, antracit, 40x21 cm	21	21	40.1
4	400003	KLIPSK	39.9	Tava mic-dejun, gri	56	36	26.1
5	400004	EKET	320.3	Corp cu 2 usi si 1 polita, gri inchis, 70x35x70 cm	70	35	70.2
6	400005	BILLY	249.45	Biblioteca, alb, 80x28x202 cm	80	28	202.1
7	400006	KALLAX	749.2	Etajera cu 4 organizatoare, negru-maro, 147x147 cm	147	39	147.1
8	400007	FINNBY	229.9	Biblioteca, negru, 60x180 cm	60	24	180.2
9	400008	MARKUS	599.99	Scaun rotativ, Vissle gri inchis	62	60	140.1
10	400009	BERGMUND	349.75	Scaun, negru/Gunnared gri mediu	52	59	96.1
11	400010	SKOGSTA/INGOLF	3433.5	Masa+6 scaune, acacia/negru, 235x100 cm	235	100	74.5
12	400011	TARSELE	2299.2	Masa extensibila, furnir stejar/negru, 150/200x80 cm	200	80	77.3
13	400012	MITTCIRKEL/ALEX	739.99	Birou, aspect pin intens/alb, 140x60 cm	140	60	73.2
14	400013	BRIMNES	799.15	Dulap cu 3 usi, negru, 117x190 cm	117	50	190.2
15	400014	KNOXHULT	1889.25	Bucatarie, alb, 220x61x220 cm	220	61	220.1
16	400015	METOD/MAXIMERA	419.55	Corp baza plita+cuptor+sertar, alb/Bodarp gri-verde, 60x60 cm	60	61	88.1
17	400016	ASPINGE	2388.15	Chicineta, negru/frasin, 120x60x202 cm	120	60	202.1
18	400017	LANDSKRONA	2499.15	Canapea 3 locuri, Gunnared verde deschis/lemn	204	89	78.3
19	400018	LANGARYD	6499.99	Canapea 3 locuri+sezlong, dreapta, Lejde gri/negru/lemn	280	135	70.5
20	400019	KIVIK	1449.55	Canapea 2 locuri, Tibbleby bej/gri	190	95	83.4
21	400020	SODERHAMN	1700.1	Sezlong, Viarp bej/maro	93	151	83.1
22	400021	ASKHOLMEN	149.35	Masa pentru perete, exterior, Pliant maro inchis, 70x44 cm	70	44	71.1
23	400022	NAMMARO	449.99	Banca cu spatar, exterior, maro deschis vopsit	62	78	50.1
24	400023	DJUNGELSKOG	129.9	Jucarie de plus, urs brun	100	50	50.1
25	400024	SATSUMAS	249.99	Suport plante, bambus/alb, 70 cm	84	28	70.1

GARANTIE	LINK_WEB
1 4 ani, prelungire 1 an	https://www.ikea.com/ro/ro/p/malm-pat-cu-depozitare-alb-20404806/
2 3 ani, fara prelungire	https://www.ikea.com/ro/ro/p/idanaes-pat-tapitat-cu-depozitare-gunnared-gri-inchis-40458964/
3 (null)	https://www.ikea.com/ro/ro/p/enhet-polita-rotativa-antracit-20465734/
4 (null)	https://www.ikea.com/ro/ro/p/klipsk-tava-mic-dejun-gri-10327700/
5 2 ani, fara prelungire	https://www.ikea.com/ro/ro/p/eket-corp-cu-2-usi-si-1-polita-gri-inchis-20344921/
6 1 an, prelungire 1 an	https://www.ikea.com/ro/ro/p/billy-biblioteca-alb-00263850/
7 2 ani, fara prelungire	https://www.ikea.com/ro/ro/p/kallax-etajera-cu-4-organizatoare-negru-maro-s09017483/
8 2 ani, fara prelungire	https://www.ikea.com/ro/ro/p/finnby-biblioteca-negru-10261129/
9 10 ani	https://www.ikea.com/ro/ro/p/markus-scaun-rotativ-vissle-gri-inchis-70261150/
10 (null)	https://www.ikea.com/ro/ro/p/bergmund-scaun-negru-gunnared-gri-mediu-s69384307/
11 5 ani, prelungire 1 an	https://www.ikea.com/ro/ro/p/skogsta-ingolf-masa-6scaune-acacia-negru-s09482693/
12 3 ani, fara prelungire	https://www.ikea.com/ro/ro/p/tarsele-masa-extensibila-furnir-stejar-negru-70581359/
13 2 ani, prelungire 1 an	https://www.ikea.com/ro/ro/p/mittcirkel-alex-birou-aspect-pin-intens-alb-s09521722/
14 2 ani, fara prelungire	https://www.ikea.com/ro/ro/p/brimnes-dulap-cu-3-usi-negru-60407577/
15 7 ani garantie, fara prelungire	https://www.ikea.com/ro/ro/p/knochhult-bucatarie-alb-s49180467/
16 (null)	https://www.ikea.com/ro/ro/p/metod-maximera-corp-baza-plita-cuptor-sertar-alb-bodarp-gri-verde-sl
17 10 ani, fara prelungire	https://www.ikea.com/ro/ro/p/aespingle-chicineta-negru-frasin-s99478168/
18 3 ani, prelungire 2 ani	https://www.ikea.com/ro/ro/p/landskrona-canapea-3-locuri-gunnared-verde-deschis-lemn-s39270326/
19 5 ani, prelungire 2 ani	https://www.ikea.com/ro/ro/p/langaryd-canapea-3locuri-sezlong-dreapta-lejde-gri-negru-lemn-s19418
20 4 ani, fara prelungire	https://www.ikea.com/ro/ro/p/kivik-canapea-2-locuri-tibbleby-bej-gri-s09440599/
21 1 an, fara prelungire	https://www.ikea.com/ro/ro/p/soederhamn-sezlong-viarp-bej-maro-s89305620/
22 (null)	https://www.ikea.com/ro/ro/p/askholmen-masa-pentru-perete-exterior-pliant-maro-inchis-70557496/
23 (null)	https://www.ikea.com/ro/ro/p/naemmaroe-banca-cu-spatar-exterior-maro-deschis-vopsit-30510302/
24 (null)	https://www.ikea.com/ro/ro/p/djungelskog-jucarie-de-plus-urs-brun-00402813/
25 (null)	https://www.ikea.com/ro/ro/p/satsumas-suport-plante-bambus-alb-90258156/

	NUME_DESIGNER	ID_CATEGORIE
1 :o/p/malm-pat-cu-depositare-alb-20404806/	Eva Lilja Lowenhielm	PASDU
2 :o/p/idanaes-pat-tapitat-cu-depositare-gunnared-gri-inchis-40458964/	Francis Cayouette	PASDU
3 :o/p/enhet-polita-rotativa-antracit-20465734/	IKEA of Sweden/E Lilja Lowenhielm	ASDRT
4 :o/p/klipsk-tava-mic-dejun-gri-10327700/	Marcus Arvonen	ASDRT
5 :o/p/eket-corp-cu-2-usi-si-l-polita-gri-inchis-20344921/	Jon Karlsson	GHJeR
6 :o/p/billy-biblioteca-alb-00263850/	Gillis Lundgren	GHJeR
7 :o/p/kallax-etajera-cu-4-organizatoare-negru-maro-s09017483/	Tord Bjorklund	GHJeR
8 :o/p/finnby-biblioteca-negru-10261129/	IKEA of Sweden	GHJeR
9 :o/p/markus-scaun-rotativ-vissle-gri-inchis-70261150/	Henrik Preutz	SDaRK
10 :o/p/bergmund-scaun-negru-gunnared-gri-mediu-s69384307/	IKEA of Sweden/K Hagberg/M Hagberg	SDsRK
11 :o/p/skogsta-ingolf-masa-6scaune-acacia-negru-s09482693/	(null)	SDsRK
12 :o/p/tarsele-masa-extensibila-furnir-stejar-negru-70581359/	(null)	SDsRK
13 :o/p/mittcirkel-alex-birou-aspect-pin-intens-alb-s09521722/	(null)	SDsRK
14 :o/p/brimnes-dulap-cu-3-usi-negru-60407577/	K Hagberg/M Hagberg	GHJeR
15 :o/p/knochhult-bucatarie-alb-s49180467/	IKEA of Sweden/Mikael Wernhammar	DFGhT
16 :o/p/metod-maximera-corp-baza-plita-cuptor-sertar-alb-bodarp-gri-verde-s19306816/	(null)	DFGhT
17 :o/p/aespingle-chicineta-negru-frasin-s99478168/	(null)	DFGhT
18 :o/p/landskrona-canapea-3-locuri-gunnared-verde-deschis-lemn-s39270326/	IKEA of Sweden/Tord Björklund	CghUI
19 :o/p/langaryd-canapea-3locuri-sezlong-dreapta-lejde-gri-negru-lemn-s19418734/	(null)	CghUI
20 :o/p/kivik-canapea-2-locuri-tibbleby-bej-gri-s09440599/	(null)	CghUI
21 :o/p/soederhamn-sezlong-viarp-bej-maro-s89305620/	Ola Wihlborg	CghUI
22 :o/p/askholmen-masa-pentru-perete-exterior-pliant-maro-inchis-70557496/	Jon Karlsson	madYU
23 :o/p/naemmaroe-banca-cu-spatar-exterior-maro-deschis-vopsit-30510302/	Nike Karlsson	madYU
24 :o/p/djungelskog-jucarie-de-plus-urs-brun-00402813/	Annie Hulden	kkRTy
25 :o/p/satsumas-suport-plante-bambus-alb-90258156/	Carl Hagerling	eedaT

Tabela ADAUGA_COMANDA

ID_PRODUS	ID_COMANDA	CANTITATE	MOMENT_TIMP
1	400023	1000000008	1 28-MAY-24 10:20:00
2	400002	1000000008	1 28-MAY-24 10:20:03
3	400005	1000000008	2 28-MAY-24 10:20:04
4	400009	1000000010	1 28-MAY-24 10:20:06
5	400010	1000000014	1 28-MAY-24 10:20:08
6	400015	1000000014	1 28-MAY-24 10:20:09
7	400006	1000000022	1 28-MAY-24 10:20:11
8	400017	1000000022	2 28-MAY-24 10:20:12
9	400013	1000000022	3 28-MAY-24 10:20:14
10	400012	1000000026	1 28-MAY-24 10:20:16
11	400009	1000000026	4 28-MAY-24 10:20:17
12	400008	1000000026	2 28-MAY-24 10:20:20
13	400007	1000000026	5 28-MAY-24 10:20:21
14	400009	1000000000	1 24-MAY-23 10:37:10
15	400012	1000000000	1 24-MAY-23 10:40:30
16	400003	1000000002	2 05-DEC-21 21:15:20
17	400011	1000000002	1 05-DEC-21 21:51:30
18	400001	1000000004	1 18-APR-24 19:22:35
19	400004	1000000004	3 18-APR-24 20:00:10
20	400016	1000000006	1 11-FEB-24 12:06:10
21	400014	1000000012	3 14-MAY-24 17:09:45
22	400007	1000000016	5 30-AUG-21 01:45:35
23	400018	1000000018	3 19-NOV-23 18:07:10
24	400023	1000000020	3 21-JUN-23 16:05:11
25	400022	1000000020	2 21-JUN-23 16:35:50

26	400021	1000000020	1 21-JUN-23 17:40:30
27	400008	1000000024	2 09-FEB-24 09:23:30
28	400020	1000000028	3 13-MAY-18 10:09:45
29	400005	1000000028	1 13-MAY-18 11:09:45
30	400024	1000000030	1 13-MAY-18 07:19:46
31	400015	1000000032	3 12-JUN-19 10:20:09
32	400006	1000000032	2 12-JUN-19 12:20:11
33	400009	1000000034	4 09-JUN-20 09:40:09
34	400021	1000000034	1 09-JUN-20 11:45:12
35	400014	1000000036	1 19-OCT-21 08:20:09
36	400022	1000000036	2 19-OCT-21 12:20:11
37	400001	1000000038	1 12-OCT-20 07:20:09
38	400023	1000000038	1 12-OCT-20 09:20:11
39	400014	1000000040	1 23-OCT-20 10:50:12
40	400003	1000000042	3 27-OCT-22 12:27:11
41	400021	1000000044	1 29-OCT-24 10:10:50

Tabela OFERTA

	ID_PRODUS	DATA_INCEPUT	DATA_SFARSIT	DISCOUNT
1	400002	28-MAY-24	01-JUN-24	24.99
2	400010	28-MAY-24	02-JUN-24	12.99
3	400007	28-MAY-24	07-JUN-24	29.99
4	400012	23-MAY-23	29-MAY-24	39.99
5	400011	01-DEC-21	10-DEC-21	34.99
6	400018	19-NOV-23	19-NOV-23	49.99
7	400007	30-AUG-21	10-SEP-21	19.99
8	400003	04-DEC-21	10-DEC-21	20

Tabela FURNIZOR

	ID_FURNIZOR	NUME	TELEFON
1	ADS	Mircea Oprea	+40794267037
2	AER3	Mihnea Vasilescu	+40712347037
3	67RT	Adriana Georgescu	+40794234567
4	0003	Petru Coman	+40791234567
5	1S	Daniel Popara	+40778967037

Tabela MATERIE_PRIMA

	ID_MATERIAL	TIP_MATERIAL	UNITATE_MASURA	PRET_UNITATE	ID_FURNIZOR
1	5000000000	MDF	metru patrat	20	0003
2	5000000003	MDF	centimetru patrat	5	1S
3	5000000006	Plastic	metru cub	2	AER3
4	5000000009	PAL	metru patrat	8	1S
5	5000000012	Vopsea	decimetru cub	3	1S
6	5000000015	Lemn masiv	metru patrat	51	67RT
7	5000000018	Spuma poliuretanica	centimetru cub	12	1S
8	5000000021	Otel galvanizat	kilogram	7	1S
9	5000000024	Hartie laminata	kilogram	3	ADS
10	5000000027	Lac acrilic	gram	1	0003
11	5000000030	Otel galvanizat	kilogram	5	ADS
12	5000000033	Poliester	kilogram	10	67RT
13	5000000036	Bumbac	centimetru cub	5	ADS

Tabela PRODUSA_DIN

	ID_PRODUS	ID_MATERIAL	UNITATI
1	400000	5000000003	6
2	400000	5000000006	3
3	400000	5000000009	10
4	400000	5000000012	1
5	400001	5000000003	6
6	400001	5000000009	5
7	400001	5000000015	3
8	400001	5000000018	1
9	400001	5000000030	4
10	400002	5000000021	2
11	400008	5000000033	3
12	400008	5000000018	5
13	400008	5000000006	2
14	400003	5000000006	1
15	400004	5000000000	4
16	400004	5000000024	1
17	400004	5000000009	3
18	400004	5000000006	6
19	400011	5000000027	2
20	400017	5000000030	3
21	400017	5000000033	6
22	400017	5000000036	2
23	400007	5000000009	5
24	400021	5000000003	14
25	400024	5000000021	2

Tabela STOC

	ID_STOC	DATA_APROVIZIONARE
1	AGFJK	01-DEC-21
2	DFGTY	14-MAY-24
3	BSJUY	18-OCT-22
4	RLJFG	12-AUG-19
5	OUNDS	15-DEC-24
6	AGFJK	13-APR-22
7	RLJFG	02-MAR-19
8	DFGTY	29-MAY-24
9	OUNDS	01-JUN-24
10	BSJUY	03-JUN-24
11	DFGTY	24-MAY-24
12	AGFJK	06-DEC-21
13	DFGTY	07-SEP-21
14	OUNDS	09-DEC-21

Tabela APROVIZIONEAZA

	ID_MAGAZIN	ID_PRODUS	ID_STOC	DATA_APROVIZIONARE	CANTITATE
1	56GH	400009	AGFJK	01-DEC-21	5
2	10UP	400015	BSJUY	18-OCT-22	21
3	10UP	400005	BSJUY	18-OCT-22	137
4	09OP	400009	RLJFG	12-AUG-19	99
5	234Y	400021	RLJFG	12-AUG-19	23
6	45E	400010	RLJFG	12-AUG-19	4
7	10UP	400003	OUNDS	15-DEC-24	78
8	234Y	400017	OUNDS	15-DEC-24	35
9	09OP	400019	AGFJK	13-APR-22	35
10	4DF	400023	AGFJK	13-APR-22	27
11	7TY6	400017	RLJFG	02-MAR-19	14
12	56GH	400000	BSJUY	18-OCT-22	67
13	56GH	400002	DFGTY	29-MAY-24	3
14	09OP	400010	OUNDS	01-JUN-24	10
15	234Y	400007	BSJUY	03-JUN-24	25
16	45E	400012	DFGTY	24-MAY-24	1
17	10UP	400011	AGFJK	06-DEC-21	34
18	234Y	400007	DFGTY	07-SEP-21	35
19	09OP	400003	OUNDS	09-DEC-21	68

Adăugare tabela DETALII_PRODUSE pentru exercițiul 10 (pe baza datelor din tabelele din DB)

```
create table detalii_produse as
select
    ac.id_produs,
    p.nume,-- join cu piesa_mobilier doar pentru asta
    sum(ac.cantitate) as cantitate_ceruta_produc,
    max(ac.moment_timp) as timp_plasare,
    case
        when (a.cantitate - sum(ac.cantitate)) < 0
        then 0
```

```

        else (a.cantitate - sum(ac.cantitate))
    end as cantitate_stoc_produs,
    a.id_stoc,
    a.data_aprovizionare as data_ultim_stoc
from aprovizioneaza a
join adauga_comanda ac on(ac.id_produs = a.id_produs)
join piesa_mobilier p on(p.id_produs = a.id_produs)
group by ac.id_produs, p.nume, a.cantitate, a.data_aprovizionare, a.id_stoc;

commit;

```

Table DETALII_PRODUSE created.

Commit complete.

Exercițiul 6

Cerință

Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent care să utilizeze toate cele 3 tipuri de colecții studiate. Apelați subprogramul.

Să se afișeze detaliiile materialelor și a dimensiunilor unui produs pentru fiecare categorie de produse și fiecare comandă care îl conține, fiind afișate doar primele 3 materiale mai scumpe (cumpărate de la furnizor) pentru produsele din acea categorie.

Implementare

```

-- concluzie: nu pot sa fac parametrii cu in/out pentru record-uri, pt ca
trebuie tipurile asta declarate local, devenind de fapt un subprogram local
in loc de un subprogram independent
-- create type pentru local in cadrul pachetelor
create or replace procedure detalii_mobila_categorie
(
    var_numere_categ in categorie.nume%type,
    -- var_rez la nivel de nr de materiale gasite (0, 1, 2 sau 3) si de
    -- comenzi in care se afla produsul
    var_rez in out integer-- numarul de rezultate intoarse, are sens sa fie
    -- parametru in out daca as avea mai multe apelari ale procedurii pentru
    -- categorii diferite si as vrea sa vad numarul de rezultate din toate
    -- categoriile insumate
)
is
    -- tablou indexat care retine produsele din fiecare categorie
    type lista_piese_mobilier is table of piesa_mobilier.nume%type index by
    binary_integer;

```

```

t_piese_mobilier lista_piese_mobilier;

-- sectiune productie piesa mobilier
type info_productie is record (
    q produsa_din.unitati%type,
    material materie_prima.tip_material%type
);
-- varray cu detaliile de productie (dpdv al materiei prime) (q=unitati,
material=tip_material)
type lista_materiale is varray(3) of info_productie;
v_materiale lista_materiale := lista_materiale();
-- tablou indexat cu toate materialele care pot sa fie retrieved
type temp_materiale is table of info_productie index by binary_integer;
t_lista_m temp_materiale;

-- sectiune comenzi
-- tablou imbricat coresp comenziilor in care au fost achizitionate piesele
de mobilier
type lista_comenzi is table of comanda.id_comanda%type;
t_comenzi lista_comenzi;

-- sectiune detalii dimensiuni piesa mobilier
type dim_mobila is record (-- asemenea unui obiect
    dim1_lungime piesa_mobilier.lungime%type,
    dim2_latime piesa_mobilier.latime%type,
    dim3_inaltime piesa_mobilier.inaltime%type
);
var_dimensiuni dim_mobila;
begin
    select p.nume
    bulk collect into t_piese_mobilier-- numele pieselor de mobilier din
categoria selectata
        from piesa_mobilier p
        join categorie c on(c.id_categoria = p.id_categoria)
        where c.nume = var_nume_categ;

    dbms_output.put_line('Categoria '||var_nume_categ||' are urmatoarele piese
de mobilier:');
    for i in 1..t_piese_mobilier.count loop
        dbms_output.put_line(' '||t_piese_mobilier(i)||' ');
        -- pt fiecare produs ii reprezint dimensiunea si adaug detalii
referitoare la primele 3 materiale
        --pas 1: dimensiuni
        select lungime, latime, inaltime
        into var_dimensiuni.dim1_lungime, var_dimensiuni.dim2_latime,
var_dimensiuni.dim3_inaltime
            from piesa_mobilier
            where piesa_mobilier.nume = t_piese_mobilier(i);

```

```

        dbms_output.put_line('      lungime:
'||var_dimensiuni.dim1_lungime||', latime: '||var_dimensiuni.dim2_latime||',
inaltime: '||var_dimensiuni.dim3_inaltime);

--pas 2: materiale
select pd.unitati, m.tip_material
bulk collect into t_lista_m
from materie_prima m
join produsa_din pd on(m.id_material = pd.id_material)-- nu e nevoie
de join si cu piesa_mobilier
    and pd.id_produs = ( select id_produs-- subcerere necorelata pentru
corespondenta la produsul curent din tabloul indexat
                    from piesa_mobilier
                    where nume = t_piese_mobilier(i))
order by m.pret_unitate desc;
/*for j in 1..t_lista_m.count loop
    dbms_output.put_line(t_lista_m(j).q||' '||t_lista_m(j).material);
end loop;*/-- arat clar ca pot sa fie mai multe materiale decat 3
v_materiale.delete;-- e nevoie sa fie sterse toate elementele, altfel
raman materialele de la produsul anterior
if t_lista_m.count > 0
then
    for j in 1..least(v_materiale.limit, t_lista_m.count) loop--
ma asigur ca nu se intrece dimensiunea maxima a varray-ului
        v_materiale.extend;
        v_materiale(j).material := t_lista_m(j).material;
        v_materiale(j).q := t_lista_m(j).q;
    end loop;
end if;
var_rez := var_rez + v_materiale.count;-- nr de materiale dintr-o
piesa de mobilier (0, 1, 2 sau 3 maxim)
dbms_output.put_line(' Detaliile primelor 3 materiale cele mai scumpe
(de la furnizor):');
if v_materiale.count > 0
then
    for j in 1..v_materiale.count loop
        dbms_output.put_line('          tip material: ' ||
v_materiale(j).material||', cantitate: '||v_materiale(j).q);
    end loop;
else
    dbms_output.put_line('  Produsul nu are specificatii legate de
materiale');
end if;

-- toate comenzile care contin produsul
select c.id_comanda
bulk collect into t_comenzi
from comanda c

```

```

join adauga_comanda ac on(c.id_comanda = ac.id_comanda)
join piesa_mobilier p on(p.id_produs = ac.id_produs)
where p.nume = t_piese_mobilier(i);-- coresp
var_rez := var_rez + t_comenzi.count;-- nr de comenzi care includ
piesa de mobilier (0 sau mai multe)
dbms_output.put_line(' Comenzi care contin piesa
'||lower(t_piese_mobilier(i))||': ');
if t_comenzi.count > 0
then
    for j in 1..t_comenzi.count loop
        dbms_output.put('   id_comanda: '||t_comenzi(j));
        dbms_output.put_line('');-- spatiere in plus
    end loop;
else
    dbms_output.put_line(' Produsul nu face parte din nicio
comanda');
end if;

end loop;

end;
/
declare
    var_total_rez integer := 0;
begin
    detalii_mobila_categorie('Rafturi, dulapuri si unitati de depozitare',
var_total_rez);
    dbms_output.put_line('Totalul rezultatelor la nivel de numar de materiale
si comenzi pentru categoria `Rafturi, dulapuri si unitati de depozitare`:');
    dbms_output.put_line(var_total_rez);
    detalii_mobila_categorie('Canapele si fotolii', var_total_rez);
    dbms_output.put_line('Totalul rezultatelor la nivel de numar de materiale
si comenzi pentru categoria `Canapele si fotolii`:');
    dbms_output.put_line(var_total_rez);
end;
/

```

The screenshot shows the Oracle SQL Developer interface. On the left, there's a tree view of tables under 'Tables (Filtered)'. In the center, a 'Worksheet' tab is open with PL/SQL code for a procedure named 'DETALII_MOBILA_CATEGORIE'. The code includes a cursor loop to handle multiple rows. Below the worksheet is a 'Script Output' window showing the results of the compilation and execution.

```

    begin
        detalii_mobila_categorie('Canapele si dulapuri');
        dbms_output.put_line('Totalul rezultat este: ');
        dbms_output.put_line(var_total_rez);
    end;

    /*verificare
    create or replace procedure produs
    (
        var_nume_piesa in piesa_mobilier.nume%type,
        var_rez in out integer
    )
    as
    begin
        select count(*) into var_rez
        from piesa_mobilier
        where nume = var_nume_piesa;
    end;
    */

```

Procedure DETALII_MOBILA_CATEGORIE compiled
PL/SQL procedure successfully completed.

DBMS OUTPUT

The screenshot shows the 'DBMS_OUTPUT' window. It displays the output of a query that retrieves information about furniture categories and their details. The output includes the name of the category ('Rafturi, dulapuri si unitati de depozitare'), its dimensions (lungime: 70, latime: 35, inaltime: 70), the top three most expensive materials (MDF, PAL, Hartie laminata), and a specific order (id_comanda: 1000000004) that contains a piece from this category.

```

Categoria Rafturi, dulapuri si unitati de depozitare are urmatoarele piese
de mobilier:
EKET
    lungime: 70, latime: 35, inaltime: 70
    Detaliile primelor 3 materiale cele mai scumpe (de la furnizor):
        tip material: MDF, cantitate: 4
        tip material: PAL, cantitate: 3
        tip material: Hartie laminata, cantitate: 1
    Comenzi care contin piesa eket:
        id_comanda: 1000000004
        .....

```

Categoria Rafturi, dulapuri si unitati de depozitare are urmatoarele piese de mobilier:

EKET
 lungime: 70, latime: 35, inaltime: 70
 Detaliile primelor 3 materiale cele mai scumpe (de la furnizor):
 tip material: MDF, cantitate: 4
 tip material: PAL, cantitate: 3
 tip material: Hartie laminata, cantitate: 1
 Comenzi care contin piesa eket:
 id_comanda: 1000000004

BILLY

lungime: 80, latime: 28, inaltime: 202
 Detaliile primelor 3 materiale cele mai scumpe (de la furnizor):
 Produsul nu are specificatii legate de materiale

Comenzi care contin piesa billy:

id_comanda: 1000000008
 id_comanda: 1000000028

KALLAX

lungime: 147, latime: 39, inaltime: 147
 Detaliile primelor 3 materiale cele mai scumpe (de la furnizor):
 Produsul nu are specificatii legate de materiale
 Comenzi care contin piesa kallax:

```
    id_comanda: 1000000022
    id_comanda: 1000000032
FINNBY
    lungime: 60, latime: 24, inaltime: 180
    Detaliile primelor 3 materiale cele mai scumpe (de la furnizor):
        tip material: PAL, cantitate: 5
    Comenzi care contin piesa finnby:
        id_comanda: 1000000016
        id_comanda: 1000000026
BRIMNES
    lungime: 117, latime: 50, inaltime: 190
    Detaliile primelor 3 materiale cele mai scumpe (de la furnizor):
        Produsul nu are specificatii legate de materiale
    Comenzi care contin piesa brimnes:
        id_comanda: 1000000022
Totalul rezultatelor la nivel de numar de materiale si comenzi pentru
categoria `Rafturi, dulapuri si unitati de depozitare`:
12
Categoria Canapele si fotolii are urmatoarele piese de mobilier:
LANDSKRONA
    lungime: 204, latime: 89, inaltime: 78
    Detaliile primelor 3 materiale cele mai scumpe (de la furnizor):
        tip material: Poliester, cantitate: 6
        tip material: Bumbac, cantitate: 2
        tip material: Otel galvanizat, cantitate: 3
    Comenzi care contin piesa landskrona:
        id_comanda: 1000000022
LANGARYD
    lungime: 280, latime: 135, inaltime: 70
    Detaliile primelor 3 materiale cele mai scumpe (de la furnizor):
        Produsul nu are specificatii legate de materiale
    Comenzi care contin piesa langaryd:
        id_comanda: 1000000018
KIVIK
    lungime: 190, latime: 95, inaltime: 83
    Detaliile primelor 3 materiale cele mai scumpe (de la furnizor):
        Produsul nu are specificatii legate de materiale
    Comenzi care contin piesa kivik:
        Produsul nu face parte din nicio comanda
SODERHAMN
    lungime: 93, latime: 151, inaltime: 83
    Detaliile primelor 3 materiale cele mai scumpe (de la furnizor):
        Produsul nu are specificatii legate de materiale
    Comenzi care contin piesa soderhamn:
        id_comanda: 1000000028
Totalul rezultatelor la nivel de numar de materiale si comenzi pentru
categoria `Canapele si fotolii`:
18
```

Exercițiu 7

Cerință

Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent care să utilizeze 2 tipuri diferite de cursoare studiate, unul dintre acestea fiind cursor parametrizat, dependent de celălalt cursor. Apelați subprogramul.

Să se afișeze numele magazinelor, împreună cu data aprovisionării a căror piese de mobilier dintr-o listă dată se află la ofertă în anul 2024/2021 și sunt valabile în stocul magazinului respectiv.

Implementare

```
-- variabila globală
create or replace type lista_piese_mobilier as varray(15) of number(6,0);
/
show errors
create or replace type coduri_locatii_table is table of number(6,0);
/
show errors
create or replace procedure magazine_cu_piese_mobilier_la_oferta
(
    v_id_piese_mobilier_cautate in lista_piese_mobilier,
    t_locatii out coduri_locatii_table
)
is
```

```

-- cursoarele vor fi pentru magazin si piesa mobilier
var_id_produs piesa_mobilier.id_produs%type;
var_nume_piesa_mobilier piesa_mobilier.nume%type;
-- cursor dinamic
type piesa_mobilier_rec is record (
    id_produs piesa_mobilier.id_produs%type,
    nume piesa_mobilier.nume%type
);
type c is ref cursor return piesa_mobilier_rec;
c_produse c;

type info_magazin_oferta is record (
    cod_magazin magazin.id_magazin%type,
    id_stoc stoc.id_stoc%type,
    marfa_adusa stoc.data_aprovizionare%type,
    oras adresa.oras%type,
    locatie adresa.strada%type,
    oferta_inceput oferta.data_inceput%type,
    oferta_sfarsit oferta.data_sfarsit%type
);
var_magazin info_magazin_oferta;
-- cursor parametrizat (dependent de cursorul c_produse)
cursor c_magazine(id piesa_mobilier.id_produs%type) is
    select distinct m.id_magazin, s.id_stoc, ap.data_aprovizionare,
a.oras, a.strada, o.data_inceput, o.data_sfarsit
    from piesa_mobilier p
    join oferta o on(o.id_produs = p.id_produs)
    join aprovizioneaza ap on(ap.id_produs = p.id_produs)
    join magazin m on(m.id_magazin = ap.id_magazin)
    join stoc s on(s.id_stoc = ap.id_stoc and ap.data_aprovizionare =
s.data_aprovizionare)
    join adresa a on(a.cod_postal = m.cod_postal)
    where ap.id_produs = id
    and ap.cantitate != 0 -- eventual la un trigger cand se adauga in
comanda poate deveni 0 (in plus ap.cantitate nu va fi niciodata null)
    and (to_char(ap.data_aprovizionare, 'YYYY') = 2024 or
to_char(ap.data_aprovizionare, 'YYYY') = 2021)
    and to_char(ap.data_aprovizionare, 'YYYY') = to_char(o.data_inceput,
'YYYY')
    and s.data_aprovizionare >= o.data_inceput and s.data_aprovizionare <=
o.data_sfarsit;

piesa_mobilier_gasita_magazin boolean; -- doar pentru o afisare corecta in
cazul in care piesa de mobilier nu e in oferta si in stoc la niciun magazin

contor integer := 1;
cod_adresa.cod_postal%type;
cod_gasit boolean;

```

```

begin
    t_locatii := coduri_locatii_table();
    open c_produse for
        select id_produs, nume
        from piesa_mobilier
        where id_produs in ( select * from table(v_id_piese_mobilier_cautate)
);
    loop
        fetch c_produse into var_id_produs, var_nume_piesa_mobilier;
        exit when c_produse%notfound;-- atunci cand select into nu mai
returneaza niciun rand

        piesa_mobilier_gasita_magazin := false;
        dbms_output.put_line('Piesa de mobilier
'||var_nume_piesa_mobilier||' e disponibila la oferta in stoc la
magazinele:');
        -- deschid cursorul parametrizat cu param din primul cursor
        open c_magazine(var_id_produs);
        loop
            fetch c_magazine into var_magazin;
            exit when c_magazine%notfound;

            piesa_mobilier_gasita_magazin := true;
            dbms_output.put_line(' ANUNT
'||to_char(var_magazin.marfa_adusa, 'YYYY'));
            dbms_output.put_line('      id_magazin:
'||var_magazin.cod_magazin||', id_stoc: '||var_magazin.id_stoc||', data
aprovizionare: '||var_magazin.marfa_adusa||', locatie: '||var_magazin.oras||'
'||var_magazin.locatie);
            dbms_output.put_line('      data la care a inceput oferta:
'||var_magazin.oferta_inceput);
            dbms_output.put_line('      data la care s-a sfarsit/ se va
sfarsi oferta: '||var_magazin.oferta_sfarsit);

            select cod_postal
            into cod
            from adresa
            where oras = var_magazin.oras and strada =
var_magazin.locatie;

            cod_gasit := false;
            for i in 1..t_locatii.count loop
                if t_locatii(i) = cod then
                    cod_gasit := true;-- daca codul exista deja in
t_locatii nu mai e adaugat (vreau coduri distincte)
                    exit;-- break
                end if;

```

```

        end loop;
        if cod_gasit = false-- altfel se adauga la t_locatii
            then
                t_locatii.extend();
                t_locatii(contor) := cod;
                contor := contor + 1;
            end if;
        end loop;

        if piesa_mobilier_gasita_magazin = false
            then
                dbms_output.put_line('Piesa de mobilier
'||var_num_piesa_mobilier||' nu e in oferta si in stoc la niciun magazin');
            end if;

            close c_magazine;
        end loop;

        close c_produse;

end;
/
declare
    t_coduri_locatii coduri_locatii_table;
    v_lista lista_piese_mobilier :=
lista_piese_mobilier(400002,400010,400007,400012,400011,400007,400003);
    contor integer;
begin
    magazine_cu_piese_mobilier_la_oferta(v_lista, t_coduri_locatii);
    dbms_output.put_line('Id-urile tuturor locatiilor distincte gasite sunt:
');
    if t_coduri_locatii.count > 0
        then
            for contor in 1..t_coduri_locatii.count loop
                dbms_output.put_line(t_coduri_locatii(contor));
            end loop;
        else
            dbms_output.put_line('Nu a fost gasita nicio locatie');
        end if;
end;
/

```

The screenshot shows the Oracle SQL Developer interface. On the left, a tree view displays the schema structure under 'Tables (Filtered)'. The main area is a 'Worksheet' tab showing a PL/SQL block:

```

    then
        for contor in 1..t_coduri_locatii.count loop
            dbms_output.put_line(t_coduri_locatii(contor));
        end loop;
    else
        dbms_output.put_line('Nu a fost gasita nicio locatie');
    end if;
end;

```

Below the worksheet is a 'Script Output' window with the following log:

- Type LISTA_PIESE_MOBILIER compiled
- Type CODURI_LOCATII_TABLE compiled
- Procedure MAGAZINE CU_PIESE_MOBILIER LA_OFERTA compiled
- PL/SQL procedure successfully completed.

DBMS OUTPUT

The screenshot shows the 'Dbms Output' window with a buffer size of 20000. It contains several entries of DBMS_OUTPUT messages from different database sessions:

- Piesa de mobilier ENHET e disponibila la oferta in stoc la magazinul: ANUNT 2024
 - id_magazin: 56GH, id_stoc: DFGTY, data aprovisionare: 28-MAY-24
 - data la care a inceput oferta: 28-MAY-24
 - data la care s-a sfarsit/ se va sfarsi oferta: 01-JUN-24
- Piesa de mobilier KLIPSK e disponibila la oferta in stoc la magazinul: ANUNT 2021
 - Piesa de mobilier ENHET e disponibila la oferta in stoc la magazinele: ANUNT 2024
 - id_magazin: 56GH, id_stoc: DFGTY, data aprovisionare: 29-MAY-24,
 - locatie: Brasov Str. Crisului, nr. 18
 - data la care a inceput oferta: 28-MAY-24
 - data la care s-a sfarsit/ se va sfarsi oferta: 01-JUN-24
 - Piesa de mobilier KLIPSK e disponibila la oferta in stoc la magazinele: ANUNT 2021
 - id_magazin: 09OP, id_stoc: OUNDS, data aprovisionare: 09-DEC-21,
 - locatie: Timisoara Bulevardul Petre Tutea, nr. 2A
 - data la care a inceput oferta: 04-DEC-21
 - data la care s-a sfarsit/ se va sfarsi oferta: 10-DEC-21
 - Piesa de mobilier FINNBY e disponibila la oferta in stoc la magazinele: ANUNT 2024
 - id_magazin: 234Y, id_stoc: BSJUY, data aprovisionare: 03-JUN-24,
 - locatie: Sibiu Calea Surii Mari 43
 - data la care a inceput oferta: 28-MAY-24
 - data la care s-a sfarsit/ se va sfarsi oferta: 07-JUN-24
 - Piesa de mobilier FINNBY e disponibila la oferta in stoc la magazinele: ANUNT 2021
 - id_magazin: 234Y, id_stoc: DFGTY, data aprovisionare: 07-SEP-21,
 - locatie: Sibiu Calea Surii Mari 43

```

        data la care a inceput oferta: 30-AUG-21
        data la care s-a sfarsit/ se va sfarsi oferta: 10-SEP-21
Piesa de mobilier SKOGSTA/INGOLF e disponibila la oferta in stoc la
magazinele:
ANUNT 2024
    id_magazin: 09OP, id_stoc: OUNDS, data aprovisionare: 01-JUN-24,
locatie: Timisoara Bulevardul Petre Tutea, nr. 2A
        data la care a inceput oferta: 28-MAY-24
        data la care s-a sfarsit/ se va sfarsi oferta: 02-JUN-24
Piesa de mobilier TARSELE e disponibila la oferta in stoc la magazinele:
ANUNT 2021
    id_magazin: 10UP, id_stoc: AGFJK, data aprovisionare: 06-DEC-21,
locatie: Constanta Str. Cumpenei, nr. 2
        data la care a inceput oferta: 01-DEC-21
        data la care s-a sfarsit/ se va sfarsi oferta: 10-DEC-21
Piesa de mobilier MITTCIRKEL/ALEX e disponibila la oferta in stoc la
magazinele:
Piesa de mobilier MITTCIRKEL/ALEX nu e in oferta si in stoc la niciun
magazin
Id-urile tuturor locatiilor distincte gasite sunt:
500238
307160
550088
900147

```

Exercițiu 8

Cerință

Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent de tip funcție care să utilizeze într-o singură comandă SQL 3 dintre tabelele create. Tratați toate excepțiile care pot apărea, inclusiv excepțiile predefinite NO_DATA_FOUND și TOO_MANY_ROWS. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate.

Să se afișeze toate piesele de mobilier care au cel puțin o materie primă asociată, fiind dată o listă de comenzi (pot exista comenzi pentru care ar putea să dea no_data_found la produse având materiale asociate, no_data_found dacă există produse fără materiale asociate în comanda curentă verificată din lista de comenzi) și pentru piesele de mobilier găsite (ca având materiale asociate în comenzi indicate) să se afișeze angajatul care a procesat comanda (doresc exact un rezultat de acest tip - too_many_rows/no_data_found).

Implementare

```

create or replace function verifica_procesare_piese_mobilier
(
    var_id_comanda_cautata in comanda.id_comanda%type
) return varchar

```

```

is
  var_msg varchar2(5000);-- ce returneaza functia (detalii comenzi cu cel
  putin un produs care are materia prima asociata)

  var_id_comanda comanda.id_comanda%type;
  var_nume_agent agent_vanzari.nume%type;

  var_id_produs piesa_mobilier.id_produs%type;
  var_nume_piesa_mobilier piesa_mobilier.nume%type;
  var_piesa_mobilier_material_gasita boolean := false;
  cursor c_produse(id comanda.id_comanda%type) is-- piesele de mobilier
  asociate cu materii prime pentru comanda curenta
    select p.id_produs, p.nume
    from piesa_mobilier p
    join adauga_comanda ac on(ac.id_produs = p.id_produs)
    where ac.id_comanda = id-- coresp parametru cursor parametrizat
    and exists (-- subcerere corelata
      select 1
      from produsa_din pd
      where pd.id_produs = p.id_produs
    );
  no_materials_associated_for_order exception;-- exceptie definita de mine,
rescriu no_data_found deoarece poate sa apara si la agenti
begin
  select id_comanda
  into var_id_comanda
  from comanda
  where id_comanda = var_id_comanda_cautata;

  var_msg := var_msg||'Comanda'||var_id_comanda||' are urmatoarele piese de
mobilier:'||chr(10);
  open c_produse(var_id_comanda);
  loop
    fetch c_produse into var_id_produs, var_nume_piesa_mobilier;
    exit when c_produse%notfound;
    var_piesa_mobilier_material_gasita := true;
    var_msg := var_msg||'  Procesare validata produse...'||chr(10);
    var_msg := var_msg||'  Produsul'||var_nume_piesa_mobilier||' are
cel putin un material asociat'||chr(10);
    begin
      select a.nume
      into var_nume_agent
      from agent_vanzari a
      join comanda c on(c.id_angajat = a.id_angajat)
      join adauga_comanda ac on(ac.id_comanda = c.id_comanda)
      where ac.id_produs = var_id_produs;
    end;
  end loop;
end;

```

```

        var_msg := var_msg||'    Procesare validata pentru exact un agent
de vanzari care a prelucrat produsul
'||var_nume_piesa_mobilier||'...'||chr(10);
        var_msg := var_msg||'        Produsul a fost procesat (intr-o
comanda sau mai multe) de un singur angajat pe nume
'||var_nume_agent||chr(10);
        var_msg := var_msg||'        Piesa: '||var_nume_piesa_mobilier||' -
Procesata de: '||var_nume_agent||chr(10);
exception
when no_data_found then
        var_msg := var_msg||'        Exceptie no_data_found: Nu au
fost gasiti angajati care sa fi procesat produsul
'||var_nume_piesa_mobilier||chr(10);
when too_many_rows then
        var_msg := var_msg||'        Exceptie too_many_rows: Prea multi
angajati care au procesat piesa de mobilier in doua sau mai multe
comenzi'||chr(10);
        var_msg := var_msg||'        Rezolvare exceptie: se ia doar un
singur angajat'||chr(10);
        select nume
        into var_nume_agent
        from (
            select a.nume
            from agent_vanzari a
            join comanda c on(c.id_angajat = a.id_angajat)
            join adauga_comanda ac on(ac.id_comanda =
c.id_comanda)
            where ac.id_produs = var_id_produs
            and rownum = 1
        );
        var_msg := var_msg||'        Produsul a fost procesat (intr-o
comanda sau mai multe) de un singur angajat pe nume
'||var_nume_agent||chr(10);
    end;
end loop;

if not var_piesa_mobilier_material_gasita
then
    raise no_materials_associated_for_order;
end if;

return var_msg;

exception
when no_data_found then
    var_msg := 'Exceptie no_data_found: Niciun rezultat intors, no_data_found
general';
    return var_msg;
when no_materials_associated_for_order then

```

```

        var_msg := var_msg || 'Exceptie no_materials_associated_for_order: Nicio
piesa de mobilier din comanda nu are asociata cel putin o materie prima';
        return var_msg;
    when others then
        var_msg := var_msg || 'Exceptie necunoscuta: ' || sqlerrm;
        return var_msg;
end;
/
declare
    v_id comanda.id_comanda%type;
begin
    v_id := 1000000008;
    -- 1000000008 ramane pt ca nu intalneste nicio exceptie (produsul 400002
ENHET care are cel putin o materie asociata face parte dintr-o singura comanda
deci e prelucrata de un singur agent)
    dbms_output.put_line(verifica_procesare_piese_mobilier(v_id));
    dbms_output.put_line('');
    v_id := 1000000000;
    -- 1000000000 ramane pt a arata exceptia no_materials (produsele 400009 si
400012 nu au asociate materiale)(de mentionat ca are produsul 400009 in comun
cu 1000000010 si 1000000026 si produsul 400009 cu 10000000026)
    dbms_output.put_line(verifica_procesare_piese_mobilier(v_id));
    dbms_output.put_line('');
    v_id := 1000000030;
    -- 1000000030 ramane pentru a arata exceptia no_data_found (are un singur
produs 400024 care are material asociat si care nu se mai regaseste in nicio
alta comanda in afara de cea curenta deci daca comanda curenta nu e prelucrata
de un angajat inseamna ca produsul nu e prelucrat de nimeni)
    dbms_output.put_line(verifica_procesare_piese_mobilier(v_id));
    dbms_output.put_line('');
    v_id := 1000000026;
    /* 1000000026 ramane pt a arata exceptia too_many_rows (produsul care are
cel putin un material asociat 400007 si face parte din comenzile 1000000026 si
1000000016, e prelucrat de angajatii 10000(comanda 1000000026) si
10030(comanda 1000000016))
    de retinut ca la 1000000026 produsul 400008 FINNBY (celalalt care are
materie asociata) e in comun cu o alta comanda dar comanda 1000000024 are
angajatul null, deci pentru produsul acesta nu va intampina nicio exceptie */
    dbms_output.put_line(verifica_procesare_piese_mobilier(v_id));
    dbms_output.put_line('');
end;
/

```

The screenshot shows the Oracle SQL Developer interface. On the left, there's a tree view of tables under 'Tables (Filtered)'. In the center, the 'Worksheet' tab contains a PL/SQL block:

```

dbms_output.put_line(verifica_procesare_piese_mobilier);
dbms_output.put_line('');
end;

/*
-- pas 1: asociere materie prima
select p.id_produs, p.numar
from piesa_mobilier p

```

Below the worksheet is the 'Script Output' tab, which displays the results of the execution:

```

Function VERIFICA_PROCESARE_PIESE_MOBILIER compiled

PL/SQL procedure successfully completed.

```

DBMS OUTPUT

Dbms Output

Buffer Size: 20000

SGBD_HOMEDB1-Alexandrescu

```

Comanda 1000000008 are urmatoarele piese de mobilier:
    Procesare validata produse...
        Produsul ENHET are cel putin un material asociat
    Procesare validata pentru exact un agent de vanzari care a
        Produsul a fost procesat (intr-o comanda sau mai multe
            -----

```

Comanda 1000000008 are urmatoarele piese de mobilier:
 Procesare validata produse...
 Produsul ENHET are cel putin un material asociat
 Procesare validata pentru exact un agent de vanzari care a prelucrat
 produsul ENHET...
 Produsul a fost procesat (intr-o comanda sau mai multe) de un singur
 angajat pe nume Andrei
 Piesa: ENHET - Procesata de: Andrei

Comanda 1000000000 are urmatoarele piese de mobilier:
 Exceptie no_materials_associated_for_order: Nicio piesa de mobilier din
 comanda nu are asociata cel putin o materie prima

Comanda 1000000030 are urmatoarele piese de mobilier:
 Procesare validata produse...
 Produsul SATSUMAS are cel putin un material asociat
 Exceptie no_data_found: Nu au fost gasiti angajati care sa fi
 procesat produsul SATSUMAS

Comanda 1000000026 are urmatoarele piese de mobilier:
 Procesare validata produse...
 Produsul FINNBY are cel putin un material asociat
 Exceptie too_many_rows: Prea multi angajati care au procesat piesa de
 mobilier in doua sau mai multe comenzi

```
Rezolvare exceptie: se ia doar un singur angajat
    Produsul a fost procesat (intr-o comanda sau mai multe) de un singur
angajat pe nume Mihai
    Procesare validata produse...
    Produsul MARKUS are cel putin un material asociat
    Procesare validata pentru exact un agent de vanzari care a prelucrat
produsul MARKUS...
    Produsul a fost procesat (intr-o comanda sau mai multe) de un singur
angajat pe nume Adela
    Piesa: MARKUS - Procesata de: Adela
```

Exercițiul 9

Cerință

Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent de tip procedură care să aibă minim 2 parametri și să utilizeze într-o singură comandă SQL 5 dintre tabelele create. Definiți minim 2 excepții proprii, altele decât cele predefinite la nivel de sistem. Apelați subprogramul astfel încât să evidențiați toate cazurile definite și tratate.

Să se afișeze pentru o materie primă piesele de mobilier care o conțin și pentru o cantitate achiziționată din piesa de mobilier cea mai vândută să se afișeze toate comenzi care o conțin pe acea piesă de mobilier selectată.

Implementare

```
create or replace procedure
piese_materie_asociata_durata_maxima_coresp_comanda
(
    material materie_prima.id_material%type,
    q adauga_comanda.cantitate%type
```

```

)
is
type info_materie_mobilier is record (
    id_prod piesa_mobilier.id_produs%type,
    nume_prod piesa_mobilier.nume%type,
    nume_mat materie_prima.tip_material%type,
    total_vanzari number
);
type t_materie_mobilier is table of info_materie_mobilier index by
binary_integer;
t_mobilier_detalii t_materie_mobilier;

cursor c_detalii_mobilier is
    select p.id_produs, p.nume, mp.tip_material, sum(ac.cantitate)
    from adauga_comanda ac
    join piesa_mobilier p on(p.id_produs = ac.id_produs)
    join produsa_din pd on(p.id_produs = pd.id_produs)
    join materie_prima mp on(mp.id_material = pd.id_material)
    where mp.id_material = material
    group by p.id_produs, p.nume, mp.tip_material
    order by p.nume;

-- cursor ciclu fara subcerere
cursor c_toate_comenzi(id piesa_mobilier.id_produs%type) is
    select distinct(id_comanda), cantitate
    from adauga_comanda
    where id_produs = id;

type info_clienti_comanda is record (
    cod_client client.id_client%type,
    cod_comanda comanda.id_comanda%type,
    data_achiz comanda.data_achizitie%type,
    plata tranzactie.modalitate_plata%type
);
type t_clienti_comanda is table of info_clienti_comanda index by
binary_integer;
t_comenzi_detalii t_clienti_comanda;

cursor c_detalii_comenzi(id piesa_mobilier.id_produs%type) is
    select c.id_client, co.id_comanda, co.data_achizitie,
t.modalitate_plata
    from client c
    join comanda co on(c.id_client = co.id_client)
    join tranzactie t on(t.id_comanda = co.id_comanda)
    join adauga_comanda ac on(co.id_comanda = ac.id_comanda)-- doar pentru
parametrul din cursor, cantitatea cautata si moment_timp
    where ac.id_produs = id
    and ac.cantitate = q

```

```

        group by c.id_client, co.id_comanda, co.data_achizitie,
t.modalitate_plata
        order by co.id_comanda;

-- computarea maximului de vanzari
max_total_vanzari number := 0;
id_produs_selectat piesa_mobilier.id_produs%type;

timestamp_primul_produs adauga_comanda.moment_timp%type;
timestamp_ultimul_produs adauga_comanda.moment_timp%type;
durata_plasare_produse interval day to second;

no_furniture exception;
no_orders exception;
begin
    -- detaliile pieselor de mobilier care contin materia prima
    open c_detalii_mobilier;
    fetch c_detalii_mobilier bulk collect into t_mobilier_detalii;

    dbms_output.put_line('Detalii piese mobilier ce contin materialul
'||material||': ');
    if t_mobilier_detalii.count = 0
        then
            raise no_furniture;
    end if;

    for i in 1..t_mobilier_detalii.count loop
        dbms_output.put_line(' Piesa de mobilier
'||t_mobilier_detalii(i).id_prod||' |||t_mobilier_detalii(i).nume_prod|| e
asociata materialului '||t_mobilier_detalii(i).nume_mat);
        dbms_output.put_line(' total vanzari:
'||t_mobilier_detalii(i).total_vanzari);
        for com in c_toate_comenzile(t_mobilier_detalii(i).id_prod) loop
            dbms_output.put_line('      Comanda '||com.id_comanda||'
(cantitate produs = '||com.cantitate||'): ');
            dbms_output.put_line('      Toate produsele din comanda curenta
(care sunt aprovizionate la un magazin): ');-- pot exista produse din comanda
care sa nu faca parte dintr-un triplet de forma (id_produs, id_magazin,
id_stoc), care nu vor fi afisate
            -- cursor ciclu cu o subcerere
            for prod in (select p.nume as produs, p.id_produs as id_prod,
a.id_stoc as stoc, a.id_magazin as mag
                from piesa_mobilier p
                join adauga_comanda ac on(ac.id_produs = p.id_produs)
                join comanda co on(co.id_comanda = ac.id_comanda)
                join aprovizioneaza a on(a.id_produs = p.id_produs)
                where co.id_comanda = com.id_comanda) loop

```

```

        dbms_output.put_line('          Produsul'||prod.produs||'
(id:'||prod.id_prod||'), se afla in stocul'||prod.stoc||' si magazinul
'||prod.mag);
      end loop;
    end loop;
end loop;

for i in 1..t_mobilier_detalii.count loop
  if t_mobilier_detalii(i).total_vanzari >= max_total_vanzari
    then
      max_total_vanzari := t_mobilier_detalii(i).total_vanzari;
      id_produs_selectat := t_mobilier_detalii(i).id_prod;
    end if;
end loop;

dbms_output.put_line('  Produsul selectat este'||id_produs_selectat||'
avand numarul total de vanzari de'||max_total_vanzari);

close c_detalii_mobilier;

-- id-ul produsului a fost gasit si se vor afisa comenzile (via un id ca
parametru in cursor) pentru care cantitatea specificata ca parametru in
subprogram din acel produs exista
open c_detalii_comenzi(id_produs_selectat);
fetch c_detalii_comenzi bulk collect into t_comenzi_detalii;

if t_comenzi_detalii.count = 0
  then
    raise no_orders;
end if;

dbms_output.put_line('cantitate: '||q||', t_comenzi_detalii:');
for i in 1..t_comenzi_detalii.count loop
  dbms_output.put_line('  Comanda'||t_comenzi_detalii(i).cod_comanda||'
a fost plasata de clientul cu id-ul'||t_comenzi_detalii(i).cod_client||',
fiind asociata tranzactia cu tipul de plata'||t_comenzi_detalii(i).plata);
  dbms_output.put_line('    data achizitie:
'||t_comenzi_detalii(i).data_achiz);

  select min(moment_timp), max(moment_timp), (max(moment_timp) -
min(moment_timp))
    into timestamp_primul_produs, timestamp_ultimul_produs,
durata_plasare_produse
     from adauga_comanda
    where id_comanda = t_comenzi_detalii(i).cod_comanda;

```

```

        dbms_output.put_line('      timestamp de inceput (plasarea primului
produs):'||timestamp_primul_produs||', timestamp de final (plasarea ultimului
produs):'||timestamp_ultimul_produs);
        dbms_output.put_line('      durata plasarii tuturor produselor in cos
pentru comanda curenta a fost de: ||durata_plasare_produse);
    end loop;

close c_detalii_comenzi;

exception
    when no_furniture then
        dbms_output.put_line('Nicio piesa de mobilier asociata materialului
'||material);
    when no_orders then
        dbms_output.put_line('Nicio comanda asociata produsului gasit avand
cantitatea'||q);
end;
/
begin
    piese_materie_asociata_durata_maxima_coresp_comanda(5000000012, 2);
    -- exceptia no_furniture deoarece 5000000012 nu e material care apartine
niciunui produs
    dbms_output.put_line('');
    piese_materie_asociata_durata_maxima_coresp_comanda(5000000036, 2);
    -- nicio exceptie deoarece dintre comenzile care contin 400017, doar
1000000022 are acest produs in cantitatea de 2
    dbms_output.put_line('');
    piese_materie_asociata_durata_maxima_coresp_comanda(5000000021, 1);
    -- am aratat ca pot exista mai multe produse cu acelasi material, 400002
si 400024
    -- nicio exceptie
    dbms_output.put_line('');
    piese_materie_asociata_durata_maxima_coresp_comanda(5000000030, 1);
    -- exceptia no_order deoarece dintre comenzile care contin 400017, doar
1000000022 are acest produs in cantitatea de 2
    dbms_output.put_line('');
    piese_materie_asociata_durata_maxima_coresp_comanda(5000000009, 5);
    -- nicio exceptie, am aratat ca daca 400007, produsul selectat, se gaseste
in aceeasi cantitate de 5 in 2 comenzi, ambele comenzi vor fi afisate
    dbms_output.put_line('');
end;
/

```

SGBD_HOMEDB1-Alexandrescu

Tables (Filtered)

- + ADAUGA_COMAND
- + ADRESA
- + AGENT_VANZARI
- + APROVIZIONEaza
- + CATEGORIE
- + CLIENT
- + CLIENT_FIZIC
- + CLIENT_ONLINE
- + COMANDA
- + COUNTRIES
- + DEPARTMENTS
- + DETALII_PRODUSE
- + EMPLOYEES
- + FURNIZOR
- + JOB_GRADES
- + JOB_HISTORY
- + JOBS
- + LOCATIONS
- + MAGAZIN
- + MATERIE_PRIMA
- + MEMBER
- + OFERTA
- + PIESA_MOBILIER
- + PRODUSA_DIN

Worksheet Query Builder

```

    dbms_output.put_line('');
    piese_materie_asociata_durata_maxima_coresp_comanda(5000000000);
    -- nicio exceptie, am aratat ca daca 400007, produsul selectat
    dbms_output.put_line('');
end;
/
/* 
select co.id_comanda, p.id_produs, p.nume, mp.tip_material
from adauga_comanda ac
join comanda co on(ac.id_comanda = co.id_comanda)
join piesa_mobilier p on(p.id_produs = ac.id_produs)
join produsa_din pd on(p.id_produs = pd.id_produs)
*/

```

Script Output X

Task completed in 0.313 seconds

Procedure PIESE_MATERIE_ASOCIATA_DURATA_MAXIMA_CORESP_COMANDA compiled

PL/SQL procedure successfully completed.

DBMS OUTPUT

Dbms Output x

Buffer Size: 20000

SGBD_HOMEDB1-Alexandrescu x

Detalii piese mobilier ce contin materialul 5000000012:
Nicio piesa de mobilier asociata materialului 5000000012

Detalii piese mobilier ce contin materialul 5000000036:
Piesa de mobilier 400017 LANDSKRONA e asociata materialului Bumbac
total vanzari: 2
Comanda 1000000022 (cantitate produs = 2):
Toate produsele din comanda curenta (care sunt aprovisionate la un magazin):

Produsul LANDSKRONA (id: 400017), se afla in stocul OUNDS si magazinul 234Y
Produsul LANDSKRONA (id: 400017), se afla in stocul RLJFG si magazinul 7TY6

Produsul selectat este 400017 avand numarul total de vanzari de 2
cantitate: 2, t_comenzi_detalii:

Comanda 1000000022 a fost plasata de clientul cu id-ul 10000070, fiind
asociata tranzactia cu tipul de plata card
data achizitie: 28-MAY-24
timestamp de inceput (plasarea primului produs): 28-MAY-24
10.20.11.000000 AM, timestamp de final (plasarea ultimului produs): 28-MAY-24
10.20.14.000000 AM

durata plasarii tuturor produselor in cos pentru comanda curenta a fost de: +00 00:00:03.000000

Detalii piese mobilier ce contin materialul 5000000021:

Piesa de mobilier 400002 ENHET e asociata materialului Otel galvanizat total vanzari: 1

Comanda 1000000008 (cantitate produs = 1):

Toate produsele din comanda curenta (care sunt aprovisionate la un magazin):

Produsul ENHET (id: 400002), se afla in stocul DFGTY si magazinul 56GH

Produsul BILLY (id: 400005), se afla in stocul BSJUY si magazinul 10UP

Produsul DJUNGELSKOG (id: 400023), se afla in stocul AGFJK si magazinul 4DF

Piesa de mobilier 400024 SATSUMAS e asociata materialului Otel galvanizat total vanzari: 1

Comanda 1000000030 (cantitate produs = 1):

Toate produsele din comanda curenta (care sunt aprovisionate la un magazin):

Produsul selectat este 400024 avand numarul total de vanzari de 1 cantitate: 1, t_comenzi_detalii:

Comanda 1000000030 a fost plasata de clientul cu id-ul 10000100, fiind asociata tranzactia cu tipul de plata card

data achizitie: 13-MAY-18

timestamp de inceput (plasarea primului produs): 13-MAY-18
07.19.46.000000 AM, timestamp de final (plasarea ultimului produs): 13-MAY-18 07.19.46.000000 AM

durata plasarii tuturor produselor in cos pentru comanda curenta a fost de: +00 00:00:00.000000

Detalii piese mobilier ce contin materialul 5000000030:

Piesa de mobilier 400001 IDANAS e asociata materialului Otel galvanizat total vanzari: 2

Comanda 1000000004 (cantitate produs = 1):

Toate produsele din comanda curenta (care sunt aprovisionate la un magazin):

Comanda 1000000038 (cantitate produs = 1):

Toate produsele din comanda curenta (care sunt aprovisionate la un magazin):

Produsul DJUNGELSKOG (id: 400023), se afla in stocul AGFJK si magazinul 4DF

Piesa de mobilier 400017 LANDSKRONA e asociata materialului Otel galvanizat

total vanzari: 2

Comanda 1000000022 (cantitate produs = 2):

Toate produsele din comanda curenta (care sunt aprovisionate la un magazin):

Produsul LANDSKRONA (id: 400017), se afla in stocul OUNDS si magazinul 234Y

Produsul LANDSKRONA (id: 400017), se afla in stocul RLJFG si magazinul 7TY6

Produsul selectat este 400017 avand numarul total de vanzari de 2 Nicio comanda asociata produsului gasit avand cantitatea 1

Detalii piese mobilier ce contin materialul 5000000009:

Piesa de mobilier 400004 EKET e asociata materialului PAL total vanzari: 3

Comanda 1000000004 (cantitate produs = 3):

Toate produsele din comanda curenta (care sunt aprovisionate la un magazin):

Piesa de mobilier 400007 FINNBY e asociata materialului PAL
total vanzari: 10
Comanda 1000000026 (cantitate produs = 5):
Toate produsele din comanda curenta (care sunt aprovisionate la un magazin):
Produsul FINNBY (id: 400007), se afla in stocul BSJUY si magazinul 234Y
Produsul FINNBY (id: 400007), se afla in stocul DFGTY si magazinul 234Y
Produsul BERGMUND (id: 400009), se afla in stocul RLJFG si magazinul 09OP
Produsul BERGMUND (id: 400009), se afla in stocul AGFJK si magazinul 56GH
Produsul MITTCIRKEL/ALEX (id: 400012), se afla in stocul DFGTY si magazinul 45E
Comanda 1000000016 (cantitate produs = 5):
Toate produsele din comanda curenta (care sunt aprovisionate la un magazin):
Produsul FINNBY (id: 400007), se afla in stocul BSJUY si magazinul 234Y
Produsul FINNBY (id: 400007), se afla in stocul DFGTY si magazinul 234Y
Piesa de mobilier 400001 IDANAS e asociata materialului PAL
total vanzari: 2
Comanda 1000000004 (cantitate produs = 1):
Toate produsele din comanda curenta (care sunt aprovisionate la un magazin):
Comanda 1000000038 (cantitate produs = 1):
Toate produsele din comanda curenta (care sunt aprovisionate la un magazin):
Produsul DJUNGELSKOG (id: 400023), se afla in stocul AGFJK si magazinul 4DF
Produsul selectat este 400007 avand numarul total de vanzari de 10
cantitate: 5, t_comenzi_detalii:
Comanda 1000000016 a fost plasata de clientul cu id-ul 10000020, fiind
asociata tranzactia cu tipul de plata card
data achizitie: 30-AUG-21
timestamp de inceput (plasarea primului produs): 30-AUG-21
01.45.35.000000 AM, timestamp de final (plasarea ultimului produs): 30-AUG-
21 01.45.35.000000 AM
durata plasarii tuturor produselor in cos pentru comanda curenta a
fost de: +00 00:00:00.000000
Comanda 1000000026 a fost plasata de clientul cu id-ul 10000090, fiind
asociata tranzactia cu tipul de plata card
data achizitie: 28-MAY-24
timestamp de inceput (plasarea primului produs): 28-MAY-24
10.20.16.000000 AM, timestamp de final (plasarea ultimului produs): 28-MAY-
24 10.20.21.000000 AM
durata plasarii tuturor produselor in cos pentru comanda curenta a
fost de: +00 00:00:05.000000

Idee intersecție prin interclasare abandonată exercițiul 9 (din cauză că ar fi o filtrare decât să trateze cazuri definite pe baza no_data_found)

Cerință

Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent de tip procedură care să aibă minim 2 parametri și să utilizeze într-o singură comandă SQL 5 dintre tabelele create. Definiți minim 2 excepții proprii, altele decât cele predefinite la nivel de sistem. Apelați subprogramul astfel încât să evidențiați toate cazurile definite și tratate.

Să se afișeze pentru o medie de preț și o lună dată (invalid_month) clienții care au plasat comenzi în mai puțin de o oră, mai mari decât media în luna respectivă (no_order_client_found). În plus, se vor returna și piesele de mobilier comandate, la nivelul uneia sau mai multor comenzi, inclusiv datele despre comandă, aprovizionarea ei și intervalul de timp (de la plasarea primului produs până la ultimul).

Implementare

```
create or replace procedure  
durata_mai_mica_1h_plasare_comenzi_mai_mare_prag_pret_la_luna
```

```

(
    prag_pret piesa_mobilier.pret%type,
    luna char-- format DD-MON-YYYY
)
is
    type lista_luni_valide is varray(12) of char(3);-- maxim 12 Luni, de aceea
    a avut rost sa folosesc si varray
        v_luni_valide lista_luni_valide := lista_luni_valide();
        var_luna_gasita boolean := false;

    type info_clienti_comanda is record (
        cod_client client.id_client%type,
        cod_comanda comanda.id_comanda%type,
        data_achiz comanda.data_achizitie%type,
        plata tranzactie.modalitate_plata%type,
        timestamp_primul_produs adauga_comanda.moment_timp%type,
        timestamp_ultimul_produs adauga_comanda.moment_timp%type,
        durata_plasare_produse interval day to second
    );
    -- doua tablouri indexate cu elemente de tip record
    type t_clienti_comanda is table of info_clienti_comanda index by
    binary_integer;
    t_clienti_prag t_clienti_comanda;
    t_clienti_durata t_clienti_comanda;
    t_clienti_prag_durata t_clienti_comanda;-- intersectie prin interclasare
    i integer := 1;
    j integer := 1;
    k integer := 0;
    cursor c_detalii_prag is
        select c.id_client, co.id_comanda, co.data_achizitie,
        t.modalitate_plata, min(ac.moment_timp), max(ac.moment_timp),
        (max(ac.moment_timp) - min(ac.moment_timp))
            from client c
            join comanda co on(c.id_client = co.id_client)
            join tranzactie t on(t.id_comanda = co.id_comanda)
            join adauga_comanda ac on(co.id_comanda = ac.id_comanda)
            join piesa_mobilier p on(p.id_produs = ac.id_produs)
            where to_char(co.data_achizitie, 'MON') = luna
            group by c.id_client, co.id_comanda, co.data_achizitie,
        t.modalitate_plata
            having sum(ac.cantitate * p.pret) > prag_pret
            order by co.id_comanda;-- trebuie sa fie sortate pentru interclasare
    cursor c_detalii_durata is
        select c.id_client, co.id_comanda, co.data_achizitie,
        t.modalitate_plata, min(ac.moment_timp), max(ac.moment_timp),
        (max(ac.moment_timp) - min(ac.moment_timp))
            from client c
            join comanda co on(c.id_client = co.id_client)

```

```

join tranzactie t on(t.id_comanda = co.id_comanda)
join adauga_comanda ac on(co.id_comanda = ac.id_comanda)
where to_char(co.data_achizitie, 'MON') = luna
group by c.id_client, co.id_comanda, co.data_achizitie,
t.modalitate_plata
having extract(hour from (max(ac.moment_timp) - min(ac.moment_timp))) 
= 0
order by co.id_comanda;

invalid_month exception;
no_order_above_price exception;
no_order_below_one_hour exception;
begin
  select distinct(to_char(data_achizitie, 'MON'))-- ne permitem sa cautam
comenzi in luni in care au existat vanzari, altfel nu
  bulk collect into v_luni_valide
from comanda;

for i in 1..v_luni_valide.count loop
  if v_luni_valide(i) = luna
    then
      var_luna_gasita := true;
    end if;
  end loop;

if var_luna_gasita = false
  then
    raise invalid_month;
end if;

dbms_output.put_line('Detalii comenzi cu un pret mai mare ca pragul
'||prag_pret||' si clienti, impreuna cu o durata de plasare a produselor sub o
ora, in luna '||luna);
open c_detalii_prag;
fetch c_detalii_prag bulk collect into t_clienti_prag;

if t_clienti_prag.count = 0
  then
    raise no_order_above_price;
end if;

/*dbms_output.put_line('t_clienti_prag:');
for i in 1..t_clienti_prag.count loop
  dbms_output.put_line(' Comanda '||t_clienti_prag(i).cod_comanda||' a
fost plasata de clientul cu id-ul '||t_clienti_prag(i).cod_client||', fiind
asociata tranzactia cu tipul de plata '||t_clienti_prag(i).plata);
  dbms_output.put_line('      data achizitie:
'||t_clienti_prag(i).data_achizitie);

```

```

        dbms_output.put_line('      timestamp de inceput (plasarea primului
produs):'||t_clienti_prag(i).timestamp_primul_produs||', timestamp de final
(plasarea ultimului produs):'||t_clienti_prag(i).timestamp_ultimul_produs);
        dbms_output.put_line(' Asadar durata plasarii produselor in cos a
fost de:'||t_clienti_prag(i).durata_plasare_produse);
    end Loop;*/

close c_detalii_prag;

open c_detalii_durata;
fetch c_detalii_durata bulk collect into t_clienti_durata;

if t_clienti_durata.count = 0
then
    raise no_order_below_one_hour;
end if;

/*dbms_output.put_line('t_clienti_durata:');
for i in 1..t_clienti_prag.count loop
    dbms_output.put_line(' Comanda'||t_clienti_durata(i).cod_comanda||
a fost plasata de clientul cu id-ul'||t_clienti_durata(i).cod_client||',
fiind asociata tranzactia cu tipul de plata'||t_clienti_durata(i).plata);
    dbms_output.put_line('      data achizitie:
'||t_clienti_durata(i).data_achiz);
    dbms_output.put_line('      timestamp de inceput (plasarea primului
produs):'||t_clienti_durata(i).timestamp_primul_produs||', timestamp de final
(plasarea ultimului produs):'||t_clienti_durata(i).timestamp_ultimul_produs);
    dbms_output.put_line(' Asadar durata plasarii produselor in cos a
fost de:'||t_clienti_durata(i).durata_plasare_produse);
end Loop;*/

close c_detalii_durata;

-- alg de intersectie prin interclasare
-- sigur exista cazuri in care t_clienti_prag are n elem, in timp ce
t_clienti_durata are m
while i <= t_clienti_prag.count and j <= t_clienti_durata.count loop
    if t_clienti_prag(i).cod_comanda = t_clienti_durata(j).cod_comanda
    then
        k := k + 1;
        t_clienti_prag_durata(k) := t_clienti_prag(i);-- conditie de
egalitate pt intersectie
        -- se avanseaza in ambele tablouri
        i := i + 1;
        j := j + 1;
    elsif t_clienti_prag(i).cod_comanda < t_clienti_durata(j).cod_comanda
    then
        i := i + 1;

```

```

        else
            j := j + 1;
        end if;
    end loop;

    if k = 0
        then
            raise no_data_found;
    end if;

    for i in 1..k loop
        dbms_output.put_line(' Comanda
'||t_clienti_prag_durata(i).cod_comanda||' a fost plasata de clientul cu id-ul
'||t_clienti_prag_durata(i).cod_client||', fiind asociata tranzactia cu tipul
de plata '||t_clienti_prag_durata(i).plata);
        dbms_output.put_line('      data achizitie:
'||t_clienti_prag_durata(i).data_achizitie;
        dbms_output.put_line('      timestamp de inceput (plasarea primului
produs): '||t_clienti_prag_durata(i).timestamp_primul_produs||', timestamp de
final (plasarea ultimului produs):
'||t_clienti_prag_durata(i).timestamp_ultimul_produs);
        dbms_output.put_line(' Asadar durata plasarii produselor in cos a
fost de: '||t_clienti_prag_durata(i).durata_plasare_produse);
    end loop;

exception
    when invalid_month then
        dbms_output.put_line('Exceptie invalid_month: Nu exista comenzi care
sa corespunda lunii '||luna);
    when no_order_above_price then
        dbms_output.put_line('Exceptie no_order_above_price: Niciun client nu
a plasat nicio comanda cu un pret mai mare decat pragul de '||prag_pret);
    when no_order_below_one_hour then
        dbms_output.put_line('Exceptie no_order_below_one_hour: Niciun client
nu a plasat nicio comanda avand o durata mai mica decat o ora');
    when no_data_found then
        dbms_output.put_line('Exceptie no_data_found: Niciun client nu a
plasat nicio comanda cu un pret mai mare decat pragul de '||prag_pret||' si cu
o durata mai mica decat o ora');
    when others then
        dbms_output.put_line('Exceptie necunoscuta: '||sqlerrm);
end;
/
begin
    durata_mai_mica_1h_plasare_comenzi_mai_mare_prag_pret_la_luna(500, 'MAY');
    --nicio exceptie
    dbms_output.put_line('');

```

```

durata_mai_mica_1h_plasare_comenzi_mai_mare_prag_pret_la_luna(1000,
'SEP');
-- exceptia invalid_month deoarece nu s-a plasat nicio comanda in Luna
septembrie a oricarui an
dbms_output.put_line('');
durata_mai_mica_1h_plasare_comenzi_mai_mare_prag_pret_la_luna(500, 'JUN');
-- exceptia no_order_below_one_hour, 3 comenzi in Luna iunie care nu
respecta proprietatea
dbms_output.put_line('');
durata_mai_mica_1h_plasare_comenzi_mai_mare_prag_pret_la_luna(2500,
'FEB');
-- exceptia no_order_above_price deoarece cele 2 comenzi (1000000024 si
1000000006) au preturi sub 2500 cu toate ca durata e mai mica decat o ora la
ambele
dbms_output.put_line('');
durata_mai_mica_1h_plasare_comenzi_mai_mare_prag_pret_la_luna(2300,
'OCT');
-- exceptia no_data_found
dbms_output.put_line('');
end;
/

```

```

durata_mai_mica_1h_plasare_comenzi_mai_mare_prag_pret_la_luna(2300, 'OCT');
-- exceptia no_data_found
dbms_output.put_line('');
end;

/*
int i = 1, j = 1;
while(i <= n && j <= m) {
    if(a[i] == b[j]) {
        k++;
    }
}

```

DBMS OUTPUT

```

Dbms Output | Buffer Size: 20000 |
SGBD_HOMEDB1-Alexandrescu x

Detalii comenzi cu un pret mai mare ca pragul 500 si clienti,
Comanda 1000000000 a fost plasata de clientul cu id-ul 1000
data achizitie: 24-MAY-23
timestamp de inceput (plasarea primului produs): 24-MAY-
```

Detalii comenzi cu un pret mai mare ca pragul 500 si clienti, impreuna cu o durata de plasare a produselor sub o ora, in luna MAY
Comanda 1000000000 a fost plasata de clientul cu id-ul 10000040, fiind asociata tranzactia cu tipul de plata card
data achizitie: 24-MAY-23
timestamp de inceput (plasarea primului produs): 24-MAY-23
10.37.10.000000 AM, timestamp de final (plasarea ultimului produs): 24-MAY-23 10.40.30.000000 AM
Asadar durata plasarii produselor in cos a fost de: +00 00:03:20.000000
Comanda 1000000008 a fost plasata de clientul cu id-ul 10000050, fiind asociata tranzactia cu tipul de plata cash
data achizitie: 28-MAY-24
timestamp de inceput (plasarea primului produs): 28-MAY-24
10.20.00.000000 AM, timestamp de final (plasarea ultimului produs): 28-MAY-24 10.20.04.000000 AM
Asadar durata plasarii produselor in cos a fost de: +00 00:00:04.000000
Comanda 1000000012 a fost plasata de clientul cu id-ul 10000030, fiind asociata tranzactia cu tipul de plata card
data achizitie: 14-MAY-24
timestamp de inceput (plasarea primului produs): 14-MAY-24
05.09.45.000000 PM, timestamp de final (plasarea ultimului produs): 14-MAY-24 05.09.45.000000 PM
Asadar durata plasarii produselor in cos a fost de: +00 00:00:00.000000
Comanda 1000000014 a fost plasata de clientul cu id-ul 10000000, fiind asociata tranzactia cu tipul de plata card
data achizitie: 28-MAY-24
timestamp de inceput (plasarea primului produs): 28-MAY-24
10.20.08.000000 AM, timestamp de final (plasarea ultimului produs): 28-MAY-24 10.20.09.000000 AM
Asadar durata plasarii produselor in cos a fost de: +00 00:00:01.000000
Comanda 1000000022 a fost plasata de clientul cu id-ul 10000070, fiind asociata tranzactia cu tipul de plata card
data achizitie: 28-MAY-24
timestamp de inceput (plasarea primului produs): 28-MAY-24
10.20.11.000000 AM, timestamp de final (plasarea ultimului produs): 28-MAY-24 10.20.14.000000 AM
Asadar durata plasarii produselor in cos a fost de: +00 00:00:03.000000
Comanda 1000000026 a fost plasata de clientul cu id-ul 10000090, fiind asociata tranzactia cu tipul de plata card
data achizitie: 28-MAY-24
timestamp de inceput (plasarea primului produs): 28-MAY-24
10.20.16.000000 AM, timestamp de final (plasarea ultimului produs): 28-MAY-24 10.20.21.000000 AM
Asadar durata plasarii produselor in cos a fost de: +00 00:00:05.000000

Exceptie invalid_month: Nu exista comenzi care sa corespunda lunii SEP

Detalii comenzi cu un pret mai mare ca pragul 500 si clienti, impreuna cu o durata de plasare a produselor sub o ora, in luna JUN
Exceptie no_order_below_one_hour: Niciun client nu a plasat nicio comanda avand o durata mai mica decat o ora

Detalii comenzi cu un pret mai mare ca pragul 2500 si clienti, impreuna cu o durata de plasare a produselor sub o ora, in luna FEB
Exceptie no_order_above_price: Niciun client nu a plasat nicio comanda cu un pret mai mare decat pragul de 2500

Detalii comenzi cu un pret mai mare ca pragul 2300 si clienti, impreuna cu o durata de plasare a produselor sub o ora, in luna OCT
Exceptie no_data_found: Niciun client nu a plasat nicio comanda cu un pret mai mare decat pragul de 2300 si cu o durata mai mica decat o ora

Exercițiu 10

Cerință

Definiți un trigger de tip LMD la nivel de comandă. Declanșați trigger-ul.

Să se afișeze modificările după inserarea unei linii din tabela adauga_comanda (având condiția să fie la un moment de timp mai mare decât cele introduse în tabelă până în acel moment), care preia id-ul produsului și actualizează stocul lui din tabela detalii_produse, știind că stocul va fi actualizat indiferent dacă există mai multe stocuri pentru același produs (pentru a vedea posibilitatea de aprovizionare a tuturor stocurilor care conțin produsul respectiv). În plus, se actualizează și data stocului, deoarece tocmai a fost modificată.

Implementare

```
-- Logica e buna ca declansatorul sa fie la nivel de instructiune si nu linie
pentru fiecare inregistrare
create or replace trigger modif_stoc
    after insert on adauga_comanda
declare
    var_id_produs piesa_mobilier.id_produs%type;
    var_stoc_curent detalii_produse.cantitate_stoc_produs%type;
    var_id_stoc detalii_produse.id_stoc%type;
    var_q adauga_comanda.cantitate%type;
    var_timp_plasare adauga_comanda.moment_timp%type;
    var_detalii_stoc number;
cursor c_cantitate_disp (id piesa_mobilier.id_produs%type) is
    select cantitate_stoc_produs, id_stoc-- toate produsele si cantitatea
Lor la nivel de stoc doar
```

```

        from detalii_produse
        where id_produs = id;

begin
    select a.id_produs, a.cantitate, a.moment_timp
    into var_id_produs, var_q, var_timp_plasare
    from adauga_comanda a
    join comanda co on(co.id_comanda = a.id_comanda)
    order by a.moment_timp desc, a.id_produs asc-- nu vor exista doua
inregistrari cu acelasi id_produs la aceeasi comanda, deci nu va aparea
niciodata too many results fetched
    fetch first 1 rows only;-- se proceseaza ultimul produs adaugat la
comanda, la o ultimul moment_timp inregistrat

    open c_cantitate_disp(var_id_produs);
    loop
        fetch c_cantitate_disp into var_stoc_curent, var_id_stoc;
        exit when c_cantitate_disp%notfound;

        if var_stoc_curent >= var_q
            then
                update detalii_produse
                    set cantitate_ceruta_produs = cantitate_ceruta_produs + var_q,
cantitate_stoc_produs = cantitate_stoc_produs - var_q, data_ultim_stoc =
to_char(var_timp_plasare, 'DD-MON-YYYY')
                        where id_produs = var_id_produs
                        and id_stoc = var_id_stoc;-- modific produsul din stocul
respectiv
            else
                raise_application_error(-20001, 'Stoc insuficient pentru produsul
'||var_id_produs||' in stocul'||var_id_stoc|| cu o capacitate de
'||var_stoc_curent);
            end if;

        end loop;
        close c_cantitate_disp;

end;
/
insert into COMANDA values(1000000046, 0, '29-OCT-24', null, 10000090, null);
insert into TRANZACTIE values(1000000047, 'card', 'in procesare', 1000000046);
--insert into ADAUGA_COMANDA values(400012, 1000000046, 2, '29-OCT-24
10:20:09');
--insert into ADAUGA_COMANDA values(400012, 1000000046, 1, '29-OCT-24
10:20:09');
insert into ADAUGA_COMANDA values(400009, 1000000046, 4, '29-OCT-24
10:20:09');

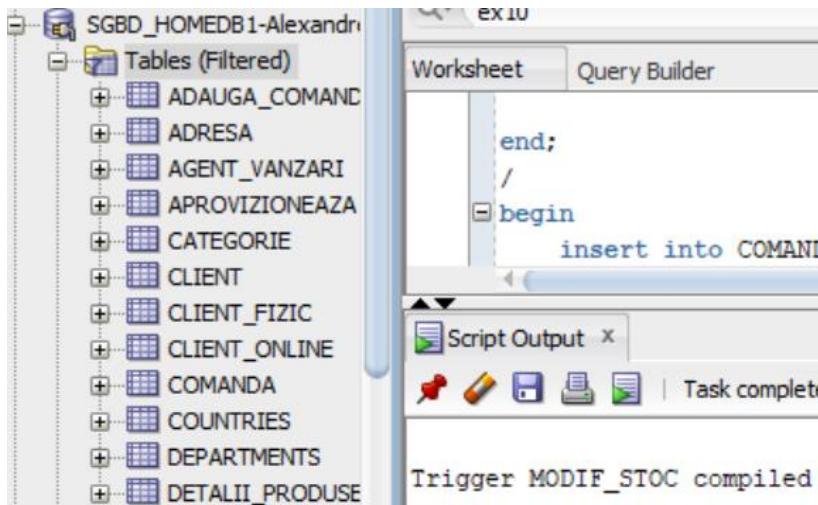
```

```

-- nu se modifica niciuna dintre cele doua inregistrari cu produsul 400009, cu
toate ca numai una nu satisface conditia,
-- iar cantitate_ceruta_produs ramane la valoarea 10, nemodificandu-se cu 14
insert into ADAUGA_COMANDA values(400023, 1000000046, 22, '29-OCT-24
10:25:09');
-- se elimina inca 22 de produse cu codul 400023 dintr-un stoc

rollback;
drop trigger modif_stoc;

```



OUTPUT

```

insert into COMANDA values(1000000046, 0, '29-OCT-24', null, 10000090, null);
insert into TRANZACTIE values(1000000047, 'card', 'in procesare', 1000000046);
--insert into ADAUGA_COMANDA values(400012, 1000000046, 2, '29-OCT-24 10:20:09');
--insert into ADAUGA_COMANDA values(400012, 1000000046, 1, '29-OCT-24 10:20:09');
insert into ADAUGA_COMANDA values(400009, 1000000046, 4, '29-OCT-24 10:20:09');
-- nu se modifica niciuna dintre cele doua inregistrari cu produsul 400009, cu toate ca numai una nu

1 row inserted.

1 row inserted.

Error starting at line : 1,546 in command -
insert into ADAUGA_COMANDA values(400009, 1000000046, 4, '29-OCT-24 10:20:09')
Error at Command Line : 1,546 Column : 13
Error report -
SQL Error: ORA-20001: Stoc insuficient pentru produsul 400009 in stocul AGFJK cu o capacitate de 0
ORA-06512: at "SGBD_HOMEDB1.MODIF_STOC", line 37
ORA-04088: error during execution of trigger 'SGBD_HOMEDB1.MODIF_STOC'

```

Screenshot of Oracle SQL Developer showing a query being run:

```
-- nu se modifica niciuna dintre cele doua inregistrari cu produsul 400009, cu toate
-- iar cantitate_ceruta_proodus ramane la valoarea 10, nemodificandu-se cu 14
insert into ADAUGA_COMANDA values(400023, 1000000046, 22, '29-OCT-24 10:25:09');

rollback;
drop trigger modif_stoc;
```

Script Output: Task completed in 0.044 seconds

1 row inserted.

DETALII_PRODUSE înainte

	ID_PRODUS	NUME	CANTITATE_CERUTA_PRODUS	TIMP_PLASARE	CANTITATE_STOC_PRODUS	ID_STOC	DATA_ULTIM_STOC
1	400005	BILLY	3	28-MAY-24 10:20:04	134	BSJUY	18-OCT-22
2	400011	TARSELE	1	05-DEC-21 21:51:30	33	AGFJK	06-DEC-21
3	400023	DJUNGELSKOG	5	28-MAY-24 10:20:00	22	AGFJK	13-APR-22
4	400009	BERGMUND	10	28-MAY-24 10:20:17	89	RLJFG	12-AUG-19
5	400003	KLIPSK	5	27-OCT-22 12:27:11	73	OUNDS	15-DEC-24
6	400009	BERGMUND	10	28-MAY-24 10:20:17	0	AGFJK	01-DEC-21
7	400007	FINNBY	10	28-MAY-24 10:20:21	15	BSJUY	03-JUN-24
8	400007	FINNBY	10	28-MAY-24 10:20:21	25	DFGTY	07-SEP-21
9	400002	ENHET	1	28-MAY-24 10:20:03	2	DFGTY	29-MAY-24
10	400010	SKOGSTA/INGOLF	1	28-MAY-24 10:20:08	9	OUNDS	01-JUN-24
11	400015	METOD/MAXIMERA	4	28-MAY-24 10:20:09	17	BSJUY	18-OCT-22
12	400012	MITTCIRKEL/ALEX	2	28-MAY-24 10:20:16	0	DFGTY	24-MAY-24
13	400003	KLIPSK	5	27-OCT-22 12:27:11	63	OUNDS	09-DEC-21
14	400010	SKOGSTA/INGOLF	1	28-MAY-24 10:20:08	3	RLJFG	12-AUG-19
15	400017	LANDSKRONA	2	28-MAY-24 10:20:12	33	OUNDS	15-DEC-24
16	400017	LANDSKRONA	2	28-MAY-24 10:20:12	12	RLJFG	02-MAR-19
17	400021	ASKHOLMEN	3	29-OCT-24 10:10:50	20	RLJFG	12-AUG-19

DETALII_PRODUSE după

	ID_PRODUS	NUME	CANTITATE_CERUTA_PRODUS	TIMP_PLASARE	CANTITATE_STOC_PRODUS	ID_STOC	DATA_ULTIM_STOC
1	400005	BILLY	3	28-MAY-24 10:20:04	134	BSJUY	18-OCT-22
2	400011	TARSELE	1	05-DEC-21 21:51:30	33	AGFJK	06-DEC-21
3	400023	DJUNGELSKOG	27	28-MAY-24 10:20:00	0	AGFJK	29-OCT-24
4	400009	BERGMUND	10	28-MAY-24 10:20:17	89	RLJFG	12-AUG-19
5	400003	KLIPSK	5	27-OCT-22 12:27:11	73	OUNDS	15-DEC-24
6	400009	BERGMUND	10	28-MAY-24 10:20:17	0	AGFJK	01-DEC-21
7	400007	FINNBY	10	28-MAY-24 10:20:21	15	BSJUY	03-JUN-24
8	400007	FINNBY	10	28-MAY-24 10:20:21	25	DFGTY	07-SEP-21
9	400002	ENHET	1	28-MAY-24 10:20:03	2	DFGTY	29-MAY-24
10	400010	SKOGSTA/INGOLF	1	28-MAY-24 10:20:08	9	OUNDS	01-JUN-24
11	400015	METOD/MAXIMERA	4	28-MAY-24 10:20:09	17	BSJUY	18-OCT-22
12	400012	MITTCIRKEL/ALEX	2	28-MAY-24 10:20:16	0	DFGTY	24-MAY-24
13	400003	KLIPSK	5	27-OCT-22 12:27:11	63	OUNDS	09-DEC-21
14	400010	SKOGSTA/INGOLF	1	28-MAY-24 10:20:08	3	RLJFG	12-AUG-19
15	400017	LANDSKRONA	2	28-MAY-24 10:20:12	33	OUNDS	15-DEC-24
16	400017	LANDSKRONA	2	28-MAY-24 10:20:12	12	RLJFG	02-MAR-19
17	400021	ASKHOLMEN	3	29-OCT-24 10:10:50	20	RLJFG	12-AUG-19

```

rollback;
drop trigger modif_stoc;

/*
select a.id_produs, a.moment_timp, a.cantitate
from adauga_comanda a

```

Script Output X | Task completed in 0.118 seconds

Rollback complete.

Trigger MODIF_STOC dropped.

Exercițiu 11

Cerință

Definiți un trigger de tip LMD la nivel de linie. Declanșați trigger-ul.

Pentru o comandă care tocmai a fost inserată în baza de date, să se modifice angajatul care o procesează numai dacă până în momentul respectiv angajatul are o productivitate mare, adică a procesat mai mult decât media de comenzi per angajat. Astfel, acesta va fi înlocuit cu un angajat care a lucrat mai puțin, adică cu cineva care a procesat un număr minim de comenzi până în acel moment (primul angajat de acest tip crescător după id-ul său).

Implementare

```

create or replace trigger update_angajat_trigger
  for insert on comanda
  compound trigger

    type info_procesari is record (
      id_comanda comanda.id_comanda%type,
      id_angajat comanda.id_angajat%type
    );
    type t_procesari is table of info_procesari index by binary_integer;
    t_proc t_procesari;
    count_proc integer := 0;
    var_nr_comenzi_procesate number;

    after each row is -- executata de fiecare data cand un rand e inserat
      -- insereaza o noua linie in t_proc, retinandu-se info despre procesarile
      obtinute
    begin
      count_proc := count_proc + 1;

```

```

t_proc(count_proc).id_comanda := :new.id_comanda;
t_proc(count_proc).id_angajat := :new.id_angajat;
end after each row;

-- se va face update la angajatii care sunt peste nivelul mediu de
productivitate, fiind inlocuiti cu altii
after statement is-- executata o sg data dupa ce toate randurile au fost
procesate
    avg_nr_comenzi_procesate number;
    min_nr_comenzi_procesate number;
begin
    -- motivul pentru care calculez media imediat dupa ce s-a adaugat
linia, e ca trebuie actualizata retinand si informatii despre noua linie
    -- similar pentru min
    select avg(nr)-- media nr de comenzi procesate de fiecare angajat
    into avg_nr_comenzi_procesate
    from ( select count(co.id_comanda) as nr
            from comanda co
            join agent_vanzari a on(co.id_angajat = a.id_angajat)-- pt a
evita comenzile cu ang null
            group by a.id_angajat );

    select min(nr)-- minimul nr de comenzi procesate de fiecare angajat
    into min_nr_comenzi_procesate
    from ( select count(co.id_comanda) as nr
            from comanda co
            join agent_vanzari a on(co.id_angajat = a.id_angajat)-- pt a
evita comenzile cu ang null
            group by a.id_angajat );

    for i in 1..t_proc.count loop-- updatez fiecare comanda cu un angajat
foarte productiv (a procesat mai mult de avg de comenzi per ang) cu unul care
a procesat un numar minim de comenzi pana in acel moment
        -- nr_comenzi_procesate trebuie sa fie calculat pentru fiecare
comanda in parte, in partea de after statement is si nu in after each row is
(iarasi apare error mutating)
        select count(co.id_comanda)-- nr de comenzi per angajat
        into var_nr_comenzi_procesate
        from comanda co
        join agent_vanzari a on(co.id_angajat = a.id_angajat)-- pt a evita
comenzile cu ang null (in cazul in care in clauza where e null, pentru ca nu a
fost asignat niciun angajat noi comenzi care nu a fost inca adaugata in baza
de date)
        where co.id_angajat = t_proc(i).id_angajat;

        if var_nr_comenzi_procesate > avg_nr_comenzi_procesate-- ang
productiv
            then

```

```

        update comanda
        set id_angajat = (
            select a.id_angajat
            from comanda co
            join agent_vanzari a on(co.id_angajat
= a.id_angajat)
            where a.id_angajat !=
group by a.id_angajat
having count(co.id_comanda) =
min_nr_comenzi_procesate-- selectez primul ang cu un numar mini de comenzi
procesate dupa id_angajat
            order by co.id_angajat
            fetch first 1 rows only )
            where id_comanda = t_proc(i).id_comanda;-- doar comanda
currenta (adica cea care tocmai a fost adaugata in baza de date)
        end if;
    end loop;
end after statement;
end;
/
delete from comanda;-- doar pt a vedea mai bine rezultatele, nu afecteaza
count()
insert into COMANDA values(1000000048, 0, '29-OCT-24', 10015, 10000090,
null);-- avg = 1/1 = 1 dar count pt 15 = 1 NU e mai mare ca avg
insert into COMANDA values(1000000050, 0, '30-NOV-24', 10015, 10000040,
null);-- avg = 2/1 = 2 dar count pt 15 = 2 NU e mai mare ca avg
insert into COMANDA values(1000000052, 0, '30-NOV-24', 10015, 10000030,
null);-- avg = 3/1 = 3 dar count pt 15 = 3 NU e mai mare ca avg
insert into COMANDA values(1000000054, 0, '30-NOV-24', 10025, 10000030,
null);-- avg = 4/2 = 2 dar count pt 25 = 1 NU e mai mare ca avg
insert into COMANDA values(1000000056, 0, '29-OCT-24', 10025, 10000090,
null);-- avg = 5/2 = 2.5 dar count pt 25 = 2 NU e mai mare ca avg
insert into COMANDA values(1000000058, 0, '29-OCT-24', 10010, 10000090,
null);-- avg = 6/3 = 2 dar count pt 10 = 1 NU e mai mare ca avg
insert into COMANDA values(1000000060, 0, '29-OCT-24', 10005, 10000090,
null);-- avg = 7/4 = 1.75 dar count pt 05 = 1 NU e mai mare ca avg
insert into COMANDA values(1000000062, 0, '29-OCT-24', 10020, 10000090,
null);-- avg = 8/5 = 1.6 dar count pt 20 = 1 NU e mai mare ca avg
insert into COMANDA values(1000000064, 0, '29-OCT-24', 10015, 10000090,
null);-- avg = 9/5 = 1.8 dar count pt 15 = 4 e mai mare ca avg
-- are de ales prima optiune dupa ordonare din (05, 10, 20) care au toate
count = 1, adica alege angajatul 10005
insert into COMANDA values(1000000066, 0, '29-OCT-24', 10025, 10000090,
null);-- avg = 10/5 = 2 dar count pt 25 = 3 e mai mare ca avg
-- are de ales prima optiune dupa ordonare din (10, 20) care au toate count =
1, adica alege angajatul 10010

```

```

insert into COMANDA values(1000000068, 0, '29-OCT-24', 10000, 10000090,
null);-- avg = 11/6 = 1.8 dar count pt 00 = 1 NU e mai mare ca avg
insert into COMANDA values(1000000070, 0, '29-OCT-24', 10030, 10000090,
null);-- avg = 12/7 = 1.7 dar count pt 30 = 1 NU e mai mare ca avg
insert into COMANDA values(1000000072, 0, '29-OCT-24', 10010, 10000090,
null);-- avg = 13/7 = 1.8 dar count pt 10 = 2 e mai mare ca avg
-- are de ales prima optiune dupa ordonare din (00, 20, 30) care au toate
count = 1, adica alege angajatul 10000
insert into COMANDA values(1000000074, 0, '29-OCT-24', 10030, 10000090,
null);-- avg = 14/7 = 2 dar count pt 30 = 2 NU e mai mare ca avg
insert into COMANDA values(1000000076, 0, '29-OCT-24', 10030, 10000090,
null);-- avg = 15/7 = 2.14 dar count pt 30 = 3 e mai mare ca avg
-- are de ales prima optiune dupa ordonare din (20) care au toate count = 1,
adica alege angajatul 10020
insert into COMANDA values(1000000078, 0, '29-OCT-24', 10025, 10000090,
null);-- avg = 16/7 = 2.28 dar count pt 25 = 4 e mai mare ca avg
-- are de ales prima optiune dupa ordonare din (00, 10, 20, 30) care au toate
count = 2, adica alege angajatul 10000

rollback;
drop trigger update_angajat_trigger;

```

```

set id_angajat = (
  select a.id_angajat
  from comanda co
  join agent_vanzari a
  where a.id_angajat != null
  group by a.id_angajat
  having count(co.id_comanda) > 1
  order by co.id_comanda
  fetch first 1 rows only
)
where id_comanda = t_proc(i).id_comanda;
end if;
end loop;
end after statement;
end;

begin
  delete from comanda;-- doar pt a vedea mai bine rezultate
  insert into COMANDA values(1000000048, 0, '29-OCT-24', 10000, 10000090);
  insert into COMANDA values(1000000050, 0, '30-NOV-24', 10000, 10000090);
end;

```

Script Output x | Task completed in 0.173 seconds

Trigger UPDATE_ANAJAT_TRIGGER compiled

OUTPUT

The screenshot shows the Oracle SQL Developer interface. On the left is a tree view of database schemas and tables. The 'Worksheet' tab is active in the main pane, displaying a series of SQL INSERT statements. Below the worksheet is a 'Script Output' window showing the results of the execution.

```
-- are de ales prima optiune dupa ordonare din (05, 10, 20) care au toate
insert into COMANDA values(1000000066, 0, '29-OCT-24', 10025, 10000090, n
-- are de ales prima optiune dupa ordonare din (10, 20) care au toate cou
insert into COMANDA values(1000000068, 0, '29-OCT-24', 10000, 10000090, n
insert into COMANDA values(1000000070, 0, '29-OCT-24', 10030, 10000090, n
insert into COMANDA values(1000000072, 0, '29-OCT-24', 10010, 10000090, n
-- are de ales prima optiune dupa ordonare din (00, 20, 30) care au toate
insert into COMANDA values(1000000074, 0, '29-OCT-24', 10030, 10000090, n
insert into COMANDA values(1000000076, 0, '29-OCT-24', 10030, 10000090, n
-- are de ales prima optiune dupa ordonare din (20) care au toate count =
insert into COMANDA values(1000000078, 0, '29-OCT-24', 10025, 10000090, n
-- are de ales prima optiune dupa ordonare din (00, 10, 20, 30) care au t

```

Script Output x | Task completed in 0.269 seconds

1 row inserted.
1 row inserted.
1 row inserted.

Tabela COMANDA înainte

	ID_COMANDA	PRET	DATA_ACHIZITIE	ID_ANGAJAT	ID_CLIENT	COD_POSTAL
1	1000000000	793.82	24-MAY-23	10020	10000040	40394
2	1000000002	1574.51	05-DEC-21	10000	10000100	(null)
3	1000000004	3460.05	18-APR-24	10000	10000080	707252
4	1000000006	2388.15	11-FEB-24	(null)	10000080	700259
5	1000000008	679.3	28-MAY-24	10010	10000050	(null)
6	1000000010	349.75	29-MAY-24	10025	10000050	(null)
7	1000000012	5667.75	14-MAY-24	(null)	10000030	106100
8	1000000014	3853.05	28-MAY-24	10025	10000000	400436
9	1000000016	919.7	30-AUG-21	10030	10000020	900147
10	1000000018	9751.92	19-NOV-23	10005	10000010	(null)
11	1000000020	1439.03	21-JUN-23	10015	10000060	(null)
12	1000000022	8144.95	28-MAY-24	10005	10000070	550088
13	1000000024	1199.98	09-FEB-24	(null)	10000090	10566
14	1000000026	4192.55	28-MAY-24	10000	10000090	10566
15	1000000028	5349.75	13-MAY-18	10030	10000040	(null)
16	1000000030	249.99	13-MAY-18	(null)	10000100	(null)
17	1000000032	0	12-JUN-19	(null)	10000030	(null)
18	1000000034	0	09-JUN-20	(null)	10000080	(null)
19	1000000036	0	19-OCT-21	(null)	10000070	(null)
20	1000000038	0	12-OCT-20	(null)	10000080	(null)
21	1000000040	0	23-OCT-20	(null)	10000040	(null)
22	1000000042	0	27-OCT-22	(null)	10000050	(null)
23	1000000044	0	29-OCT-24	(null)	10000090	(null)

Tabela COMANDA după

	ID_COMANDA	PRET	DATA_ACHIZITIE	ID_ANGAJAT	ID_CLIENT	COD_POSTAL
1	1000000048	0	29-OCT-24	10015	10000090	(null)
2	1000000050	0	30-NOV-24	10015	10000040	(null)
3	1000000052	0	30-NOV-24	10015	10000030	(null)
4	1000000054	0	30-NOV-24	10025	10000030	(null)
5	1000000056	0	29-OCT-24	10025	10000090	(null)
6	1000000058	0	29-OCT-24	10010	10000090	(null)
7	1000000060	0	29-OCT-24	10005	10000090	(null)
8	1000000062	0	29-OCT-24	10020	10000090	(null)
9	1000000064	0	29-OCT-24	10005	10000090	(null)
10	1000000066	0	29-OCT-24	10010	10000090	(null)
11	1000000068	0	29-OCT-24	10000	10000090	(null)
12	1000000070	0	29-OCT-24	10030	10000090	(null)
13	1000000072	0	29-OCT-24	10000	10000090	(null)
14	1000000074	0	29-OCT-24	10030	10000090	(null)
15	1000000076	0	29-OCT-24	10020	10000090	(null)
16	1000000078	0	29-OCT-24	10000	10000090	(null)

Worksheet Query Builder

```
-- are de ales prima optiune dupa ordonare din (00, 20, 30
insert into COMANDA values(1000000074, 0, '29-OCT-24', 100
insert into COMANDA values(1000000076, 0, '29-OCT-24', 100
-- are de ales prima optiune dupa ordonare din (20) care a
insert into COMANDA values(1000000078, 0, '29-OCT-24', 100
-- are de ales prima optiune dupa ordonare din (00, 10, 20

rollback;
drop trigger update_angajat_trigger;
```

Script Output

Task completed in 0.075 seconds

Rollback complete.

Trigger UPDATE_ANAJAT_TRIGGER dropped.

The screenshot shows the Oracle SQL Developer interface. On the left is a tree view of database schemas and tables. The central area contains a 'Worksheet' tab with a 'Query Builder' section displaying DDL code for inserting data into 'COMANDA' and dropping a trigger. Below the worksheet is a 'Script Output' window showing the results of the rollback and trigger drop operation. The bottom status bar indicates the task completed in 0.075 seconds.

Exercițiu 12

Cerință

Definiți un trigger de tip LDD. Declanșați trigger-ul.

Să se înregistreze într-un tabel ddl_log toate operațiile efectuate cu succes (doar) în BD și, dacă obiectul la care s-a dat create/drop sau alter, să îl fie afișate coloanele.

Implementare

```
create table ddl_log
(
    moment timestamp,
    user_name varchar2(30),
    ddl_event varchar2(25),
    object_name varchar2(100)
);
create or replace trigger ddl_log_trigger
    after create or alter or drop on schema
declare
    cursor c_col is
        select column_name, data_type
        from user_tab_columns
        where table_name = sys.dictionary_obj_name;-- dicționar care retine
numele obiectului alterat curent
        -- nu o să se vadă modificările imediat după ce alterez dacă adaug
coloane la tabel
    var_denumiri_coloane varchar2(1000) := '';
begin
    if sys.dictionary_obj_type = 'TABLE'
        then
            for i in c_col loop-- cursor ciclu
                var_denumiri_coloane := var_denumiri_coloane||i.column_name||'
('||i.data_type||'), ';
            end loop;
            -- în cazul în care adaug o coloană la tabelul asuora caruia
Lucrez, nu e afișata denumirea ei
            -- în mesajul dat de mine, doar dacă dau descriere va fi în regulă
            dbms_output.put_line('Coloanele din tabela
'||sys.dictionary_obj_name||' sunt : '||var_denumiri_coloane);
        end if;
    -- user
        insert into ddl_log values (systimestamp, sys_context('userenv',
'session_user'), sys.sysevent, sys.dictionary_obj_name);
end;
/
create table test
(
    id number primary key,
```

```

        nume varchar2(50),
        telefon varchar2(12),
        varsta number
);
alter table test add (descriere varchar2(100));
desc test;
alter table test drop column id;
drop table test;
create sequence seq_client-- exista cu acest nume o secv deja
start with 10000000
increment by 10
maxvalue 99999999
nocycle
nocache;
alter table aprovisioneaza modify id_stoc null;-- face parte din cheia primara
compusa

drop trigger ddl_log_trigger;
drop table ddl_log;

```

The screenshot shows the Oracle SQL Developer interface. On the left, there is a tree view of the database schema, including tables like CATEGORIE, CLIENT, and various DETALII_PRODUSE, EMPLOYEES, and MEMBER tables. The main workspace is titled 'ex12' and contains a 'Worksheet' tab with the following DDL code:

```

        end if;
        -- user
        insert into ddl_log values (systimestamp)
end;
/
create table test
(
    id number primary key,
    nume varchar2(50),
    telefon varchar2(12),
    descriere varchar2(100)
);

```

Below the worksheet is a 'Script Output' tab which displays the results of the executed commands:

```

Table DDL_LOG created.

Trigger DDL_LOG_TRIGGER compiled

```

OUTPUT

The screenshot shows the Oracle SQL Developer interface. On the left, the 'Tables (Filtered)' tree view displays various tables from the 'SGBD_HOMEDB1-Alexandru' schema, including ADAUGA_COMAND, ADRESA, AGENT_VANZARI, APROVIZIONEAZA, CATEGORIE, CLIENT, CLIENT_FIZIC, CLIENT_ONLINE, COMANDA, COUNTRIES, DDL_LOG, DEPARTMENTS, DETALII_PRODUSE, EMPLOYEES, FURNIZOR, JOB_GRADES, JOB_HISTORY, and TEST.

The main window contains two tabs: 'Worksheet' and 'Query Builder'. The 'Worksheet' tab is active, showing the results of a query:

```
Table TEST created.  
Table TEST altered.  
Name Null? Type  
----  
ID NOT NULL NUMBER  
NUME VARCHAR2(50)  
TELEFON VARCHAR2(12)  
VARSTA NUMBER  
DESCRIERE VARCHAR2(100)  
Table TEST altered.  
Table TEST dropped.
```

The 'Script Output' tab at the bottom shows the command history and errors:

```
Script Output X | Task completed in 0.206 seconds  
Table TEST created.  
Table TEST altered.  
Name Null? Type  
----  
ID NOT NULL NUMBER  
NUME VARCHAR2(50)  
TELEFON VARCHAR2(12)  
VARSTA NUMBER  
DESCRIERE VARCHAR2(100)  
Table TEST altered.  
Table TEST dropped.
```



```
Script Output X | Task completed in 1.394 seconds  
Error starting at line : 1,766 in command -  
create sequence seq_client-- exista cu acest nume o secv deja  
start with 10000000  
increment by 10  
maxvalue 99999999  
nocycle  
nocache  
Error report -  
ORA-00955: name is already used by an existing object  
00955. 00000 - "name is already used by an existing object"  
*Cause:  
*Action:  
  
Error starting at line : 1,772 in command -  
alter table aprovizioneaza modify id_stoc null  
Error report -  
ORA-01451: column to be modified to NULL cannot be modified to NULL  
01451. 00000 - "column to be modified to NULL cannot be modified to NULL"  
*Cause: the column may already allow NULL values, the NOT NULL constraint  
is part of a primary key or check constraint.  
*Action: if a primary key or check constraint is enforcing the NOT NULL  
constraint then drop that constraint
```

DBMS OUTPUT

Dbms Output × | Buffer Size: 20000 |

SGBD_HOMEDB1-Alexandrescu ×

```
Coloanele din tabela TEST sunt : ID (NUMBER), NUME (VARCHAR2), TELEFON (VARCHAR2), VARSTA (NUMBER),
Coloanele din tabela TEST sunt : ID (NUMBER), NUME (VARCHAR2), TELEFON (VARCHAR2), VARSTA (NUMBER),
Coloanele din tabela TEST sunt : NUME (VARCHAR2), TELEFON (VARCHAR2), VARSTA (NUMBER), DESCRIERE (VARCHAR2),
Coloanele din tabela TEST sunt :
```

Tabela DDL_LOG

MOMENT	USER_NAME	DDL_EVENT	OBJECT_NAME
1 01-JAN-25 22:13:30	SGBD_HOMEDB1	CREATE	SYS_C0014376
2 01-JAN-25 22:13:30	SGBD_HOMEDB1	CREATE	TEST
3 01-JAN-25 22:13:30	SGBD_HOMEDB1	ALTER	TEST
4 01-JAN-25 22:13:31	SGBD_HOMEDB1	ALTER	TEST
5 01-JAN-25 22:13:31	SGBD_HOMEDB1	DROP	TEST
6 01-JAN-25 22:16:23	SGBD_HOMEDB1	DROP	TRIGGER_LOG

ADRESA
AGENT_VANZARI
APROVIZIONEAZA
CATEGORIE
CLIENT
CLIENT_FIZIC
CLIENT_ONLINE
COMANDA
COUNTIES
DDL_LOG
DEPARTMENTS
DETALII_PRODUSE
ID_PRODUS
NUME
CANTITATE_CI
TIMP_PLASARE
CANTITATE_ST
ID_STOC
DATA_ULTIMUL_STOC
EMPLOYEES
FURNIZOR
JOB_GRADES
JOB_HISTORY

```
nocache;
alter table aprovizioneaza modify id_stoc null;--  

drop trigger ddl_log_trigger;  

drop table ddl_log;  

--ex13  

/* se considera un pachet care gestioneaza vanzari  

ce folosesc ca variabile globale cursoare si table  

un tablou va descrie materialele pieselor de mobil  

un alt tablou care ofera informatii despre comenzi  

*/  

drop package gestionare_vanzari;
```

Script Output × | Task completed in 0.076 seconds |

Trigger DDL_LOG_TRIGGER dropped.

Table DDL_LOG dropped.

Exercițiu 13

Cerință

Formulați în limbaj natural o problemă pe care să o rezolvați folosind un pachet care să includă tipuri de date complexe și obiecte necesare unui flux de acțiuni integrate, specifice bazei de date definite (minim 2 tipuri de date, minim 2 funcții, minim 2 proceduri).

Se consideră un pachet care gestionează vânzările, conținând 2 proceduri și 2 funcții, ce folosesc ca variabile globale cursori și tablouri: un tablou va descrie materialele pieselor de mobilier (tip_material, unități, preț_producție (= preț_unitate * unități)), iar un alt tablou care oferă informații despre comenziilor unor clienți dați care sunt la ofertă.

- 1) verifica_oferta_client verifică dacă un client a beneficiat de cel puțin o ofertă și returnează true sau false
- 2) calcul_total_produse_la_oferta primește o comandă și returnează numărul de produse aflate la ofertă
- 3) afiseaza_materiale_produse_la_oferta primește o comandă și, dacă comanda a beneficiat de cel puțin o ofertă (calcul_total_produse_la_oferta > 0), populează t_lista_m și afișează rezultatele (se calculează un preț unitar de vânzare, preț unitar de producție, profit firmă)
- 4) afiseaza_procent_clienti_cu_oferte selectează dintr-un cursor care conține toți clienții, populează t_comenzi (și afișează rezultatele) și se afișează din t_comenzi detaliile via afiseaza_materiale_produse_la_oferta (practic logica de a avea date la comun la mai multe funcții/ subprograme și de a apela o funcție în interiorul alteia) (și se afișează rezultatele)

Implementare

```
drop package gestionare_vanzari;
create or replace package gestionare_vanzari as
    -- 2 tipuri complexe ca variabile globale
    -- tablou indexat cu materialele pieselor de mobilier pentru procedura
    afiseaza_materiale_produse_la_oferta
        type info_productie is record (
            nume_produs piesa_mobilier.nume%type,
            pret_produs piesa_mobilier.pret%type,
            pret_productie_produs number,
            profit number
        );
        type temp_materiale is table of info_productie index by binary_integer;
        t_lista_m temp_materiale;

    -- tablou imbricat care ofera informatii despre comenzi aflate la ofertă
    -- pentru procedura afiseaza_procent_clienti_cu_oferte
        type info_comenzi_oferta is record (
            id_client client.id_client%type,
            id_comanda comanda.id_comanda%type,
            id_produs piesa_mobilier.id_produs%type,
            pret_produs piesa_mobilier.pret%type,
            cantitate adauga_comanda.cantitate%type,
            pret_neredus_total number,
            oferta_discount oferta.discount%type,
```

```

        pret_discounted_unitate number,
        pret_discounted_total number,
        oferta_inceput oferta.data_inceput%type,
        oferta_sfarsit oferta.data_sfarsit%type
    );
    type lista_comenzi is table of info_comenzi_oferta;
    t_comenzi lista_comenzi;

    function verifica_oferta_client(p_id_client in client.id_client%type)
return boolean;
    function calcul_total_produse_la_oferta(p_id_comanda in
comanda.id_comanda%type) return number;
    procedure afiseaza_materiale_produse_la_oferta(p_id_comanda in
comanda.id_comanda%type);
    procedure afiseaza_clienti_cu_oferte;

end gestionare_vanzari;
/
create or replace package body gestionare_vanzari as

    function verifica_oferta_client
    (
        p_id_client in client.id_client%type
    ) return boolean
    is
        count_comenzi_oferta number;
    begin
        select count(distinct ac.id_comanda) -- distinct e necesar altfel se
considera o comanda de mai multe ori daca sunt mai multe produse din acea
comanda la oferta
            into count_comenzi_oferta
            from adauga_comanda ac
            join comanda co on(ac.id_comanda = co.id_comanda)
            join oferta o on(ac.id_produs = o.id_produs)
            where co.data_achizitie>=o.data_inceput and
co.data_achizitie<=o.data_sfarsit
            and co.id_client = p_id_client;

        if count_comenzi_oferta > 0
            then
                return true;
        end if;
        return false;
    end verifica_oferta_client;

    function calcul_total_produse_la_oferta
    (
        p_id_comanda in comanda.id_comanda%type

```

```

) return number
is
    count_produse_oferta number;
begin
    select nvl(sum(a.cantitate), 0)
    into count_produse_oferta
    from adauga_comanda a
    join comanda co on(a.id_comanda = co.id_comanda)
    join oferta o on(a.id_produs = o.id_produs)
    where co.data_achizitie>=o.data_inceput and
co.data_achizitie<=o.data_sfarsit
        and a.id_comanda = p_id_comanda;

    return count_produse_oferta;
end calcul_total_produse_la_oferta;

procedure afiseaza_materiale_produse_la_oferta
(
    p_id_comanda in comanda.id_comanda%type
)
is
begin
    t_lista_m := temp_materiale();
    dbms_output.put_line(' Ma aflu in
afiseaza_materiale_produse_la_oferta');
    -- verific daca comanda data are sau nu produse la oferta
    if calcul_total_produse_la_oferta(p_id_comanda) > 0 then
        -- mereu se reseteaza inainte de a se popula pentru a nu retine
materialele dintr-o alta comanda (in cazul in care se apeleaza procedura de
mai mult de o data)
        t_lista_m.delete;

        select p.nume, p.pret,
sum(to_number(m.pret_unitate)*to_number(pd.unitati)), p.pret-
sum(to_number(m.pret_unitate)*to_number(pd.unitati))
        bulk collect into t_lista_m
        from piesa_mobilier p
        -- sectiune join pt productie materiale
        join produsa_din pd on(p.id_produs = pd.id_produs)
        join materie_prima m on(pd.id_material = m.id_material)
        -- sectiune join pt oferta
        join adauga_comanda a on(a.id_produs = p.id_produs)
        join oferta o on(p.id_produs = o.id_produs)
        join comanda co on(a.id_comanda = co.id_comanda)
        where co.data_achizitie>=o.data_inceput and
co.data_achizitie<=o.data_sfarsit-- oferta in vigoare
            and co.id_comanda = p_id_comanda
            group by p.nume, p.pret;

```

```

if t_lista_m.count = 0
    then
        dbms_output.put_line('      Comanda'||p_id_comanda|| are
produse la oferta, insa acestea nu au materiale asociate');
    else
        for i in 1..t_lista_m.count loop
            dbms_output.put_line('      Cu toate ca exista produsul
'||t_lista_m(i).nume_produs|| la oferta in comanda'||p_id_comanda||',
profitul firmei pentru produsul curent este'||t_lista_m(i).profit);
            dbms_output.put_line('              nume produs:
'||t_lista_m(i).nume_produs);
            dbms_output.put_line('              pret unitar produs:
'||t_lista_m(i).pret_produs||', pret productie produs:
'||t_lista_m(i).pret_productie_produs);
            dbms_output.put_line('              profit:
'||t_lista_m(i).profit);
            end loop;
        end if;

        else
            dbms_output.put_line('      Comanda'||p_id_comanda|| nu are
produse la oferta');
        end if;
end afiseaza_materiale_produse_la_oferta;

procedure afiseaza_clienti_cu_oferte
is
    cursor c_clienti is
        select id_client from client;
begin
    t_comenzi := lista_comenzi();

    for rec in c_clienti loop-- cursor ciclu
        if verifica_oferta_client(rec.id_client)
            then
                dbms_output.put_line('Clientul'||rec.id_client||' a
beneficiat de cel putin o oferta');

                select co.id_client, co.id_comanda, p.id_produs, p.pret,
a.cantitate, p.pret*a.cantitate, o.discount, round(p.pret*(100-
o.discount)/100, 2), round(p.pret*(100-o.discount)/100, 2)*a.cantitate,
o.data_inceput, o.data_sfarsit
                    bulk collect into t_comenzi
                    from comanda co
                    join adauga_comanda a on(a.id_comanda = co.id_comanda)
                    join piesa_mobilier p on(a.id_produs = p.id_produs)
                    join oferta o on p.id_produs = o.id_produs

```

```

        where co.data_achizitie>=o.data_inceput and
co.data_achizitie<=o.data_sfarsit
            and co.id_client = rec.id_client
            order by co.id_comanda, p.id_produs;

        for i in 1..t_comenzi.count loop
            if i = 1 or t_comenzi(i).id_comanda != t_comenzi(i-
1).id_comanda-- daca id-ul comenzii s-a schimbat
                then
                    dbms_output.put_line('Ma aflu in
afiseaza_clienti_cu_oferte');
                        dbms_output.put_line(' id client:
'||t_comenzi(i).id_client||', id comanda: '||t_comenzi(i).id_comanda||', id
produs: '||t_comenzi(i).id_produs';
                            dbms_output.put_line(' pret produs unitate:
'||t_comenzi(i).pret_produs||', pret produs discounted unitate:
'||t_comenzi(i).pret_discounted_unitate);
                                dbms_output.put_line(' cantitate produs in
comanda: '||t_comenzi(i).cantitate;
                                    dbms_output.put_line(' pret produs *
cantitate: '||t_comenzi(i).pret_neredus_total||', pret produs discounted *
cantitate: '||t_comenzi(i).pret_discounted_total);
                                        dbms_output.put_line(' inceput oferta:
'||t_comenzi(i).oferta_inceput||', sfarsit oferta:
'||t_comenzi(i).oferta_sfarsit);
                                            afiseaza_materiale_produse_la_oferta(t_comenzi
(i).id_comanda);-- se apeleaza procedura afiseaza_materiale_produse_la_oferta
pentru fiecare comanda distincta
                                                dbms_output.put_line('');
                                            end if;
                                        end loop;
                                    else
                                        dbms_output.put_line('Clientul '||rec.id_client||' nu a
beneficiat de nicio oferta');
                                    end if;
                                    dbms_output.put_line('');
                                    dbms_output.put_line('');
                                end loop;
                            end afiseaza_clienti_cu_oferte;

end gestionare_vanzari;
/
declare
    total_produse number;
begin
    gestionare_vanzari.afiseaza_clienti_cu_oferte;
    if gestionare_vanzari.verifica_oferta_client(10000090) then

```

```

        dbms_output.put_line('Clientul a beneficiat de cel putin o oferta');
    else
        dbms_output.put_line('Clientul nu a beneficiat de nicio oferta');
    end if;
    total_produse := gestionare_vanzari.calcul_total_produse_la_oferta(1000000026);
    dbms_output.put_line(total_produse);
    gestionare_vanzari.afiseaza_materiale_produse_la_oferta(1000000026);
end;
/

```

The screenshot shows the Oracle SQL Developer interface. On the left, there is a tree view of tables in the 'SGBD_HOMEDB1-Alexandru' schema. In the center, a 'Worksheet' tab is open with the following PL/SQL code:

```

        end loop;

    end afiseaza_clienti_cu_oferte;

end gestionare_vanzari;
/
declare
    total_produse number;
begin
    gestionare_vanzari.afiseaza_clienti_cu_oferte;
    if gestionare_vanzari.verifica_oferta_client(100000090) then

```

Below the worksheet, a 'Script Output' window shows the results of the compilation:

- Package GESTIONARE_VANZARI dropped.
- Package GESTIONARE_VANZARI compiled
- Package Body GESTIONARE_VANZARI compiled

The screenshot shows the Oracle SQL Developer interface. On the left, there is a tree view of tables in the 'SGBD_HOMEDB1-Alexandru' schema. In the center, a 'Worksheet' tab is open with the following PL/SQL code:

```

        dbms_output.put_line('Clientul a );
    else
        dbms_output.put_line('Clientul nu ');
    end if;
    total_produse := gestionare_vanzari.c;
    dbms_output.put_line(total_produse);
    gestionare_vanzari.afiseaza_materiale_
end;
/

```

Below the worksheet, a 'Script Output' window shows the results of the compilation:

- PL/SQL procedure successfully completed.

DBMS OUTPUT

The screenshot shows the Oracle SQL Developer interface with the 'Dbms Output' window open. The window title is 'Dbms Output' and the tab bar shows 'SGBD_HOMEDB1-Alexandrescu'. The buffer size is set to 20000. The output displays three client records, each detailing an offer they received.

```
Clientul 10000000 a beneficiat de cel putin o oferta
Ma aflu in afiseaza_clienti_cu_oferte
    id client: 10000000, id comanda: 1000000014, id produs:
    pret produs unitate: 3433.5, pret produs discounted unitate:
    cantitate produs in comanda: 1
    pret produs * cantitate: 3433.5, pret produs discounted * cantitate:
    inceput oferta: 28-MAY-24, sfarsit oferta: 02-JUN-24

Clientul 10000000 a beneficiat de cel putin o oferta
Ma aflu in afiseaza_clienti_cu_oferte
    id client: 10000000, id comanda: 1000000014, id produs: 400010
    pret produs unitate: 3433.5, pret produs discounted unitate: 2987.49
    cantitate produs in comanda: 1
    pret produs * cantitate: 3433.5, pret produs discounted * cantitate:
    2987.49
    inceput oferta: 28-MAY-24, sfarsit oferta: 02-JUN-24
    Ma aflu in afiseaza_materiale_produse_la_oferta
        Comanda 1000000014 are produse la oferta, insa acestea nu au
        materiale asociate

Clientul 10000010 a beneficiat de cel putin o oferta
Ma aflu in afiseaza_clienti_cu_oferte
    id client: 10000010, id comanda: 1000000018, id produs: 400018
    pret produs unitate: 6499.99, pret produs discounted unitate: 3250.64
    cantitate produs in comanda: 3
    pret produs * cantitate: 19499.97, pret produs discounted * cantitate:
    9751.92
    inceput oferta: 19-NOV-23, sfarsit oferta: 19-NOV-23
    Ma aflu in afiseaza_materiale_produse_la_oferta
        Comanda 1000000018 are produse la oferta, insa acestea nu au
        materiale asociate

Clientul 10000020 a beneficiat de cel putin o oferta
Ma aflu in afiseaza_clienti_cu_oferte
    id client: 10000020, id comanda: 1000000016, id produs: 400007
    pret produs unitate: 229.9, pret produs discounted unitate: 183.94
    cantitate produs in comanda: 5
    pret produs * cantitate: 1149.5, pret produs discounted * cantitate:
    919.7
    inceput oferta: 30-AUG-21, sfarsit oferta: 10-SEP-21
    Ma aflu in afiseaza_materiale_produse_la_oferta
```

Cu toate ca exista produsul FINNBY la oferta in comanda 1000000016, profitul firmei pentru produsul curent este 189.9
nume produs: FINNBY
pret unitar produs: 229.9, pret productie produs: 40
profit: 189.9

Clientul 10000030 nu a beneficiat de nicio oferta

Clientul 10000040 a beneficiat de cel putin o oferta
Ma aflu in afiseaza_clienti_cu_oferte
id client: 10000040, id comanda: 1000000000, id produs: 400012
pret produs unitate: 739.99, pret produs discounted unitate: 444.07
cantitate produs in comanda: 1
pret produs * cantitate: 739.99, pret produs discounted * cantitate:
444.07
inceput oferta: 23-MAY-23, sfarsit oferta: 29-MAY-24
Ma aflu in afiseaza_materiale_produse_la_oferta
Comanda 1000000000 are produse la oferta, insa acestea nu au
materiale asociate

Clientul 10000050 a beneficiat de cel putin o oferta
Ma aflu in afiseaza_clienti_cu_oferte
id client: 10000050, id comanda: 1000000008, id produs: 400002
pret produs unitate: 50.5, pret produs discounted unitate: 37.88
cantitate produs in comanda: 1
pret produs * cantitate: 50.5, pret produs discounted * cantitate: 37.88
inceput oferta: 28-MAY-24, sfarsit oferta: 01-JUN-24
Ma aflu in afiseaza_materiale_produse_la_oferta
Cu toate ca exista produsul ENHET la oferta in comanda 1000000008, profitul firmei pentru produsul curent este 36.5
nume produs: ENHET
pret unitar produs: 50.5, pret productie produs: 14
profit: 36.5

Clientul 10000060 nu a beneficiat de nicio oferta

Clientul 10000070 nu a beneficiat de nicio oferta

Clientul 10000080 nu a beneficiat de nicio oferta

Clientul 10000090 a beneficiat de cel putin o oferta
Ma aflu in afiseaza_clienti_cu_oferte
id client: 10000090, id comanda: 1000000026, id produs: 400007
pret produs unitate: 229.9, pret produs discounted unitate: 160.95
cantitate produs in comanda: 5
pret produs * cantitate: 1149.5, pret produs discounted * cantitate:
804.75
inceput oferta: 28-MAY-24, sfarsit oferta: 07-JUN-24
Ma aflu in afiseaza_materiale_produse_la_oferta
Cu toate ca exista produsul FINNBY la oferta in comanda 1000000026, profitul firmei pentru produsul curent este 189.9

```
nume produs: FINNBY
pret unitar produs: 229.9, pret productie produs: 40
profit: 189.9
```

Clientul 10000100 a beneficiat de cel putin o oferta
Ma aflu in afiseaza_clienti_cu_oferte
id client: 10000100, id comanda: 1000000002, id produs: 400003
pret produs unitate: 39.9, pret produs discounted unitate: 31.92
cantitate produs in comanda: 2
pret produs * cantitate: 79.8, pret produs discounted * cantitate: 63.84
inceput oferta: 04-DEC-21, sfarsit oferta: 10-DEC-21
Ma aflu in afiseaza_materiale_produse_la_oferta
Cu toate ca exista produsul KLIPSK la oferta in comanda 1000000002,
profitul firmei pentru produsul curent este 37.9
nume produs: KLIPSK
pret unitar produs: 39.9, pret productie produs: 2
profit: 37.9
Cu toate ca exista produsul TARSELE la oferta in comanda 1000000002,
profitul firmei pentru produsul curent este 2297.2
nume produs: TARSELE
pret unitar produs: 2299.2, pret productie produs: 2
profit: 2297.2

Clientul a beneficiat de cel putin o oferta
6
Ma aflu in afiseaza_materiale_produse_la_oferta
Cu toate ca exista produsul FINNBY la oferta in comanda 1000000026,
profitul firmei pentru produsul curent este 189.9
nume produs: FINNBY
pret unitar produs: 229.9, pret productie produs: 40
profit: 189.9