**Proiect baze de date**

Baza de date pe care am creeat-o este folositoare pentru gestionarea operațiunilor logistice în mai multe depozite. Cu ajutorul acestei baze de date putem verifica informații despre produsele aflate în fiecare depozit, care este furnizorul principal de bunuri al depozitului precum si cine este responsabilul depozitului. Modelul permite gestionarea stocurilor din fiecare depozit, depozitele și locațiile în care se află fiecare, gestionarea preturilor și cantităților din fiecare produs etc. , toate acestea pentru a facilita analiza și administrarea eficientă a unei afaceri.

Tabela **Furnizori** stochează detalii despre entitățile care furnizează bunuri către afacere, precum numele furnizorului și adresa de e-mail. Acest tabel facilitează gestionarea și comunicarea cu furnizorii. Fiecare depozit are un furnizor.

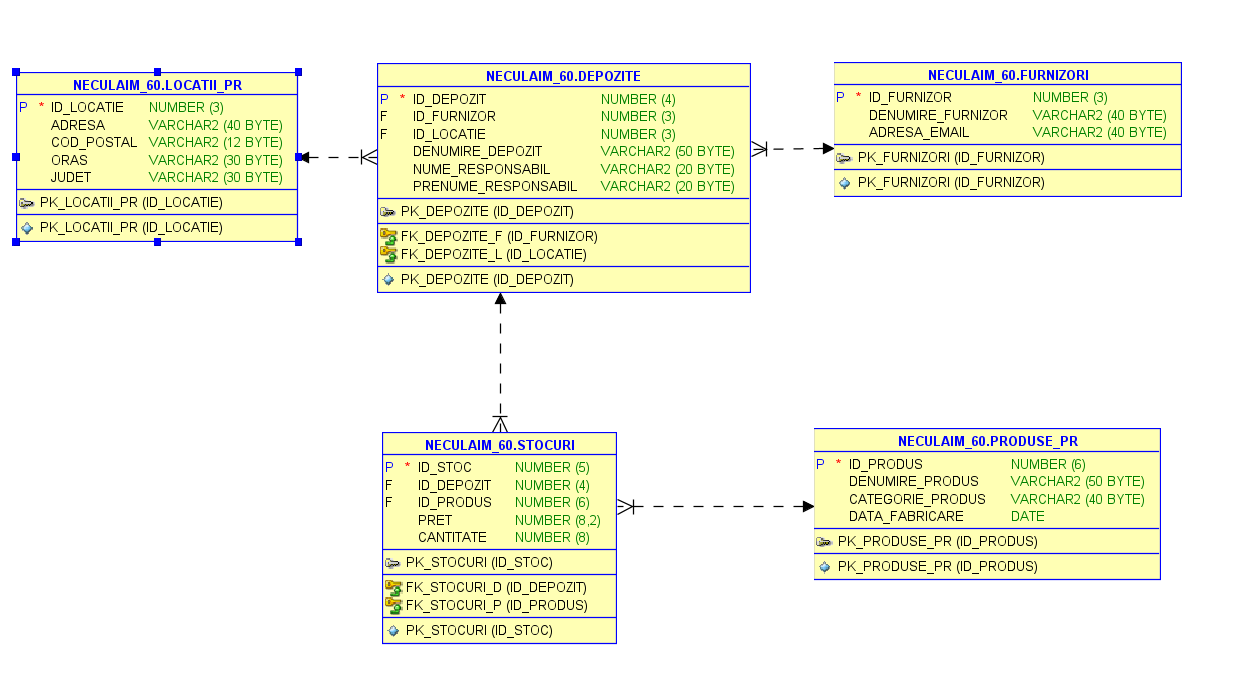
**Locatii\_pr** conține informații referitoare la locații și adrese asociate depozitelor. Aceasta tabelă permite identificarea precisa a fiecarui depozit astfel putem afla locatia fiecarui produs aflat intr-unul dintre stocuri.

Tabelul **Depozite** stochează date despre depozitele folosite pentru stocarea produselor furnizate. Fiecare depozit este asociat cu un furnizor și o locație specifică.

În **Produse\_pr** sunt înregistrate detalii despre produsele disponibile, inclusiv denumirea, categoria și data de fabricare a fiecărui produs.

**Stocuri** este tabela responsabilă pentru monitorizarea cantităților disponibile și a prețurilor produselor în fiecare depozit. Fiecare depozit poate avea mai multe stocuri iar un produs se poate afla în mai multe stocuri.

**Schema conceptuală**



**Construirea bazei de date – tabele, legături între tabele și restricții de integritate.**

**CREATE TABLE** furnizori(

id\_furnizor NUMBER(4) CONSTRAINT pk\_furnizori PRIMARY KEY,

denumire\_furnizor VARCHAR2(40),

adresa\_email VARCHAR2(10)

);

**ALTER TABLE** furnizori

**MODIFY** adresa\_email VARCHAR2(40);

**ALTER TABLE** furnizori

**MODIFY** id\_furnizor NUMBER(3);

**CREATE TABLE** locatii\_pr(

id\_locatie NUMBER(3) CONSTRAINT pk\_locatii\_pr PRIMARY KEY,

adresa VARCHAR2(40),

cod\_postal VARCHAR2(12),

oras varchar2(30),

judet varchar2(30)

);

**ALTER TABLE** locatii\_pr

**MODIFY** cod\_postal VARCHAR2(6);

**CREATE TABLE** depozite(

id\_depozit number(4) CONSTRAINT pk\_depozite PRIMARY KEY,

id\_furnizor NUMBER(3),

id\_locatie NUMBER(3),

denumire\_depozit VARCHAR2(50),

nume\_responsabil VARCHAR2(20),

prenume\_responsabil VARCHAR2(20),

CONSTRAINT fk\_depozite\_f FOREIGN KEY(id\_furnizor) REFERENCES furnizori(id\_furnizor),

CONSTRAINT fk\_depozite\_l FOREIGN KEY(id\_locatie) REFERENCES locatii\_pr(id\_locatie)

);

**ALTER TABLE** depozite

**MODIFY** denumire\_depozit VARCHAR2(50) NOT NULL;

**CREATE TABLE** produse\_pr(

id\_produs NUMBER(6) CONSTRAINT pk\_produse\_pr PRIMARY KEY,

denumire\_produs VARCHAR2(50),

categorie\_produs VARCHAR2(40),

descriere\_produs VARCHAR2(1000)

);

**ALTER** **TABLE** produse\_pr

**DROP** **COLUMN** descriere\_produs;

**CREATE TABLE** stocuri(

id\_stoc NUMBER(5) CONSTRAINT pk\_stocuri PRIMARY KEY,

id\_depozit NUMBER(4),

id\_produs NUMBER(6),

pret NUMBER(8,2),

cantitate NUMBER(8),

data\_intrare\_stoc DATE,

CONSTRAINT fk\_stocuri\_d FOREIGN KEY(id\_depozit) REFERENCES depozite(id\_depozit),

CONSTRAINT fk\_stocuri\_p FOREIGN KEY(id\_produs) REFERENCES produse\_pr(id\_produs));

**Operatii de actualizare a datelor**

**--Tabela ‘Furnizori’:**

INSERT INTO furnizori (id\_furnizor, denumire\_furnizor, adresa\_email) VALUES (100, 'ProFurnizor', 'contact@profurnizor.com');

INSERT INTO furnizori (id\_furnizor, denumire\_furnizor, adresa\_email) VALUES (101, 'Global Suppliers', 'info@globalsuppliers.com');

INSERT INTO furnizori (id\_furnizor, denumire\_furnizor, adresa\_email) VALUES (102, 'Sunrise Traders', 'sales@sunrisetraders.com');

INSERT INTO furnizori (id\_furnizor, denumire\_furnizor, adresa\_email) VALUES (103, 'Alpha Merchandise', 'support@alphamerchandise.com');

INSERT INTO furnizori (id\_furnizor, denumire\_furnizor, adresa\_email) VALUES (104, 'Market Connect', 'info@marketconnect.com');

INSERT INTO furnizori (id\_furnizor, denumire\_furnizor, adresa\_email) VALUES (105, 'Trade Masters', 'customerservice@trademasters.com');

INSERT INTO furnizori (id\_furnizor, denumire\_furnizor, adresa\_email) VALUES (106, 'Tech Solutions Co.', 'contact@techsolutionsco.com');

INSERT INTO furnizori (id\_furnizor, denumire\_furnizor, adresa\_email) VALUES (107, 'Aqua Products Inc.', 'hello@aquaproductsinc.com');

INSERT INTO furnizori (id\_furnizor, denumire\_furnizor, adresa\_email) VALUES (108, 'Furniture World', 'sales@furnitureworld.com');

INSERT INTO furnizori (id\_furnizor, denumire\_furnizor, adresa\_email) VALUES (109, 'Supply Hub', 'orders@supplyhub.com');

**Tabela ‘Locatii\_pr’:**

INSERT INTO locatii\_pr (id\_locatie, adresa, cod\_postal, oras, judet) VALUES (1, 'Strada Victoriei, Nr. 10', '100100', 'Bucuresti', 'Bucuresti');

INSERT INTO locatii\_pr (id\_locatie, adresa, cod\_postal, oras, judet) VALUES (2, 'Bulevardul Republicii, Nr. 25', '200200', 'Cluj-Napoca', 'Cluj');

INSERT INTO locatii\_pr (id\_locatie, adresa, cod\_postal, oras, judet) VALUES (3, 'Strada Unirii, Nr. 15', '300300', 'Iasi', 'Iasi');

INSERT INTO locatii\_pr (id\_locatie, adresa, cod\_postal, oras, judet) VALUES (4, 'Bulevardul Timisoara, Nr. 30', '400400', 'Timisoara', 'Timis');

INSERT INTO locatii\_pr (id\_locatie, adresa, cod\_postal, oras, judet) VALUES (5, 'Aleea Parcului, Nr. 5', '500500', 'Brasov', 'Brasov');

INSERT INTO locatii\_pr (id\_locatie, adresa, cod\_postal, oras, judet) VALUES (6, 'Strada Libertatii, Nr. 12', '600600', 'Constanta', 'Constanta');

INSERT INTO locatii\_pr (id\_locatie, adresa, cod\_postal, oras, judet) VALUES (7, 'Bulevardul Revolutiei, Nr. 7', '700700', 'Oradea', 'Bihor');

INSERT INTO locatii\_pr (id\_locatie, adresa, cod\_postal, oras, judet) VALUES (8, 'Strada Independentei, Nr. 22', '800800', 'Sibiu', 'Sibiu');

INSERT INTO locatii\_pr (id\_locatie, adresa, cod\_postal, oras, judet) VALUES (9, 'Aleea Crangului, Nr. 9', '900900', 'Ploiesti', 'Prahova');

INSERT INTO locatii\_pr (id\_locatie, adresa, cod\_postal, oras, judet) VALUES (10, 'Bulevardul Nicolae Balcescu, Nr. 18', '101010', 'Craiova', 'Dolj');

**Tabela ‘Depozite:**

INSERT INTO depozite (id\_depozit, id\_furnizor, id\_locatie, denumire\_depozit, nume\_responsabil, prenume\_responsabil) VALUES (1001, 100, 1, 'Depozit Central', 'Neculai', 'Marius-Adrian');

INSERT INTO depozite (id\_depozit, id\_furnizor, id\_locatie, denumire\_depozit, nume\_responsabil, prenume\_responsabil) VALUES (1002, 101, 2, 'Depozit Aurora', 'Ionescu', 'Mihai');

INSERT INTO depozite (id\_depozit, id\_furnizor, id\_locatie, denumire\_depozit, nume\_responsabil, prenume\_responsabil) VALUES (1003, 102, 3, 'Depozit Meridian', 'Radulescu', 'Elena');

INSERT INTO depozite (id\_depozit, id\_furnizor, id\_locatie, denumire\_depozit, nume\_responsabil, prenume\_responsabil) VALUES (1004, 103, 4, 'Depozit Vesticon', 'Georgescu', 'Andrei');

INSERT INTO depozite (id\_depozit, id\_furnizor, id\_locatie, denumire\_depozit, nume\_responsabil, prenume\_responsabil) VALUES (1005, 104, 5, 'Depozit Estwave', 'Constantinescu', 'Maria');

INSERT INTO depozite (id\_depozit, id\_furnizor, id\_locatie, denumire\_depozit, nume\_responsabil, prenume\_responsabil) VALUES (1006, 105, 6, 'Depozit Marex', 'Dumitrescu', 'Cristian');

INSERT INTO depozite (id\_depozit, id\_furnizor, id\_locatie, denumire\_depozit, nume\_responsabil, prenume\_responsabil) VALUES (1007, 106, 7, 'Depozit Microlog', 'Stoica', 'Laura');

INSERT INTO depozite (id\_depozit, id\_furnizor, id\_locatie, denumire\_depozit, nume\_responsabil, prenume\_responsabil) VALUES (1008, 107, 8, 'Depozit NeoNova', 'Florescu', 'Alex');

INSERT INTO depozite (id\_depozit, id\_furnizor, id\_locatie, denumire\_depozit, nume\_responsabil, prenume\_responsabil) VALUES (1009, 108, 9, 'Depozit RetroHome', 'Popa', 'Gabriela');

INSERT INTO depozite (id\_depozit, id\_furnizor, id\_locatie, denumire\_depozit, nume\_responsabil, prenume\_responsabil) VALUES (1010, 109, 10, 'Depozit Nexus', 'Neagu', 'Ion');

INSERT INTO depozite (id\_depozit, id\_furnizor, id\_locatie, denumire\_depozit, nume\_responsabil, prenume\_responsabil) VALUES (1011, 100, 1, 'Depozit GreenEdge', 'Marinescu', 'Andreea');

INSERT INTO depozite (id\_depozit, id\_furnizor, id\_locatie, denumire\_depozit, nume\_responsabil, prenume\_responsabil) VALUES (1012, 101, 2, 'Depozit FastTrack', 'Gheorghiu', 'Diana');

INSERT INTO depozite (id\_depozit, id\_furnizor, id\_locatie, denumire\_depozit, nume\_responsabil, prenume\_responsabil) VALUES (1013, 102, 3, 'Depozit Zenith', 'Mihai', 'Alexandru');

INSERT INTO depozite (id\_depozit, id\_furnizor, id\_locatie, denumire\_depozit, nume\_responsabil, prenume\_responsabil) VALUES (1014, 103, 4, 'Depozit Terra', 'Dragomir', 'Ana');

INSERT INTO depozite (id\_depozit, id\_furnizor, id\_locatie, denumire\_depozit, nume\_responsabil, prenume\_responsabil) VALUES (1015, 104, 5, 'Depozit Skyline', 'Vasilescu', 'Daniel');

**Tabela ‘Produse’:**

INSERT ALL

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (1, 'Cafetieră automată', 'Bucătărie')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (2, 'Boxe Bluetooth', 'Electronice')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (3, 'Fier de călcat cu aburi', 'Electrocasnice')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (4, 'Aparat de ras electric', 'Ingrijire personala')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (5, 'Ceas inteligent', 'Electronice')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (6, 'Set de unelte pentru grădinărit', 'Gradinărit')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (7, 'Aparat de aer condiționat', 'Electrocasnice')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (8, 'Set de cutite pentru bucatarie', 'Bucătărie')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (9, 'Mașină de tuns iarba', 'Gradinărit')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (10, 'Boxe stereo pentru PC', 'Electronice')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (11, 'Robot de bucătărie multifuncțional', 'Bucătărie')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (12, 'Consolă de jocuri', 'Electronice')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (13, 'Aparat de cafea espresso', 'Bucătărie')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (14, 'Husă pentru laptop', 'Accesorii')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (15, 'Lampă de birou LED', 'Iluminat')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (16, 'Prelungitor cu multiple prize', 'Accesorii')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (17, 'Aspirator portabil', 'Electrocasnice')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (18, 'Aparat de vidat alimente', 'Bucătărie')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (19, 'Smartphone performant', 'Telefoane')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (20, 'Mașină de spălat rufe', 'Electrocasnice')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (21, 'Set de prosoape premium', 'Textile')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (22, 'Tocator electric pentru bucătărie', 'Bucătărie')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (23, 'Aparat de gătit cu abur', 'Bucătărie')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (24, 'Trotinetă electrică pliabilă', 'Sport și timp liber')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (25, 'Aparat de purificat aerul', 'Electrocasnice')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (26, 'Cafetieră french press', 'Bucătărie')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (27, 'Mouse wireless ergonomic', 'Accesorii')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (28, 'Smartwatch sport', 'Accesorii')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (29, 'Set de căni termoizolante', 'Bucătărie')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (30, 'Mixer vertical pentru bucătărie', 'Bucătărie')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (31, 'Aparat de gătit cu presiune', 'Bucătărie')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (32, 'Căști audio wireless', 'Electronice')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (33, 'Filtru de cafea permanent', 'Bucătărie')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (34, 'Aparat de încălzit ceară pentru epilat', 'Ingrijire personala')

INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (35, 'Smart TV 4K', 'Electronice')

SELECT \* FROM DUAL;

**Tabela ‘Stocuri’:**

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (1, 1001, 1, 120.50, 25, TO\_DATE('2023-01-15', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (2, 1002, 2, 199.99, 15, TO\_DATE('2023-02-20', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (3, 1003, 3, 85.00, 30, TO\_DATE('2023-03-10', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (4, 1004, 4, 45.80, 20, TO\_DATE('2023-04-05', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (5, 1005, 5, 299.00, 12, TO\_DATE('2023-05-15', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (6, 1006, 6, 129.90, 18, TO\_DATE('2023-06-20', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (7, 1007, 7, 599.99, 5, TO\_DATE('2023-07-10', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (8, 1008, 8, 79.00, 28, TO\_DATE('2021-08-05', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (9, 1009, 9, 249.50, 10, TO\_DATE('2023-09-15', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (10, 1010, 10, 79.99, 22, TO\_DATE('2021-10-20', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (11, 1011, 11, 49.00, 17, TO\_DATE('2023-11-25', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (12, 1012, 12, 399.00, 8, TO\_DATE('2022-12-05', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (13, 1013, 13, 139.80, 14, TO\_DATE('2022-03-10', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (14, 1014, 14, 29.99, 32, TO\_DATE('2022-02-15', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (15, 1015, 15, 69.50, 19, TO\_DATE('2023-06-10', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (16, 1001, 16, 39.00, 40, TO\_DATE('2023-07-15', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (17, 1002, 17, 129.99, 18, TO\_DATE('2022-08-20', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (18, 1003, 18, 149.00, 25, TO\_DATE('2023-09-25', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (19, 1004, 19, 19.80, 50, TO\_DATE('2021-10-30', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (20, 1005, 20, 899.00, 7, TO\_DATE('2022-11-04', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (21, 1006, 21, 59.90, 28, TO\_DATE('2019-12-10', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (22, 1007, 22, 199.99, 15, TO\_DATE('2020-01-15', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (23, 1008, 23, 29.00, 33, TO\_DATE('2020-02-20', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (24, 1009, 24, 249.50, 11, TO\_DATE('2021-03-25', 'YYYY-MM-DD'));

INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc) VALUES (25, 1010, 25, 499.99, 6, TO\_DATE('2022-04-30', 'YYYY-MM-DD'));

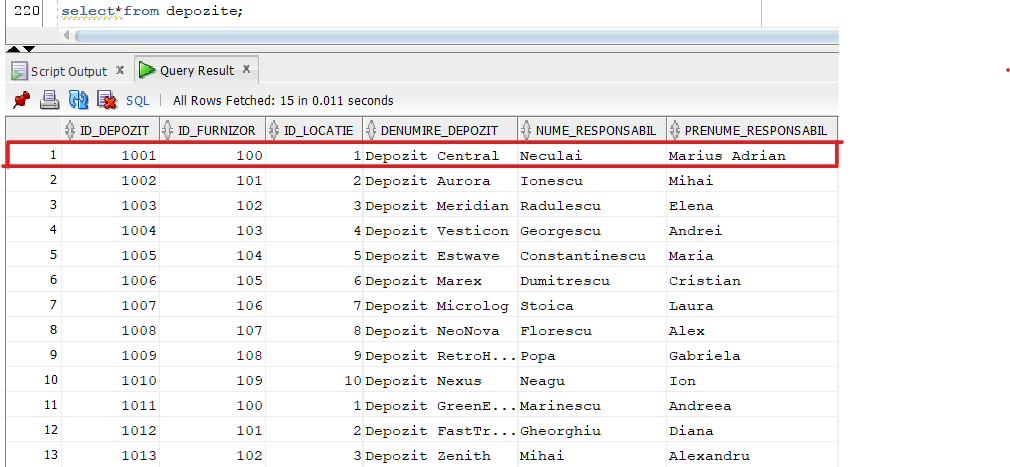
UPDATE depozite

SET prenume\_responsabil='Marius Adrian'

WHERE id\_depozit=1001;

**Inregistrarea cu numele studentului se află în tabela ‘Depozite’ care se poate selecta cu urmatoatea comanda:**

**SELECT \* From depozite where id\_depozit = 1001;**

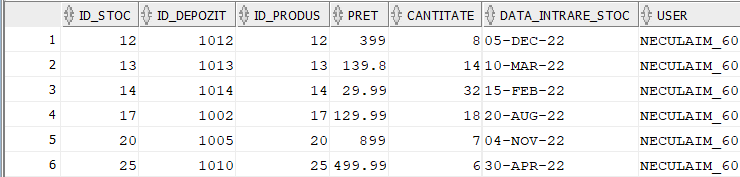
****



**Exemple de interogări**

1. Să se selecteze toate stocurile care au data de intrare în anul 2019.

SELECT \* FROM stocuri

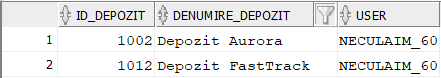
WHERE EXTRACT(YEAR FROM data\_intrare\_stoc) = 2022; 

1. Să se selecteze id-ul si denumirea depozitelor care se află in județul Cluj.

SELECT d.id\_depozit,d.denumire\_depozit FROM depozite d,locatii\_pr l

WHERE d.id\_locatie=l.id\_locatie

AND UPPER(l.judet) like 'CLUJ';



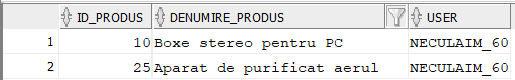
1. Să se afiseze id-ul si denumirea produselor care se află în depozitul cu denumirea “Depozit Nexus”.

SELECT p.id\_produs, p.denumire\_produs FROM produse\_pr p, stocuri s, depozite d

WHERE p.id\_produs=s.id\_produs

AND d.id\_depozit=s.id\_depozit

AND UPPER(d.denumire\_depozit) LIKE 'DEPOZIT NEXUS';

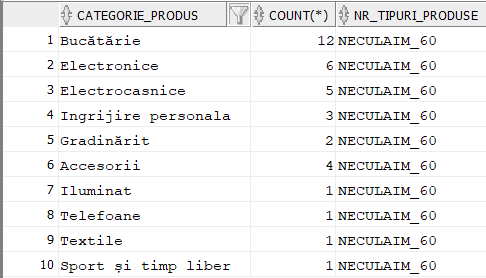


1. Să se afișeze numărul total de tipuri de produse pentru fiecare categorie de produs

SELECT categorie\_produs, COUNT(\*) AS nr\_tipuri\_produse

FROM produse\_pr

GROUP BY categorie\_produs;



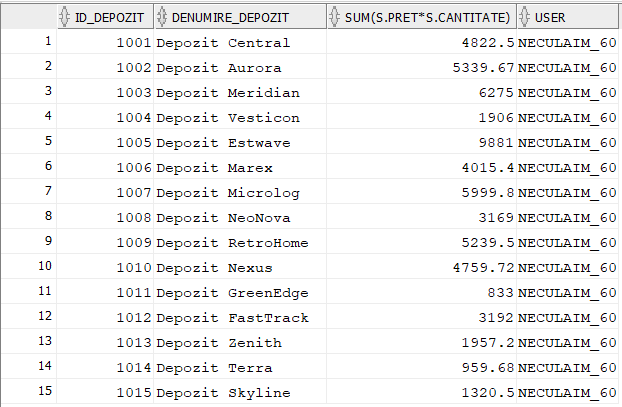
1. Să se calculeze valoarea totală a produselor aflate în fiecare depozit.

SELECT d.id\_depozit,d.denumire\_depozit,sum(s.pret\*s.cantitate) from produse\_pr p,stocuri s,depozite d

WHERE p.id\_produs=s.id\_produs

AND d.id\_depozit=s.id\_depozit

GROUP BY d.id\_depozit,d.denumire\_depozit;



1. Să se calculeze valoarea totala a produselor aflate în depozite, grupate în funcție de orașul din care fac parte.

SELECT l.oras, sum(s.pret\*s.cantitate)

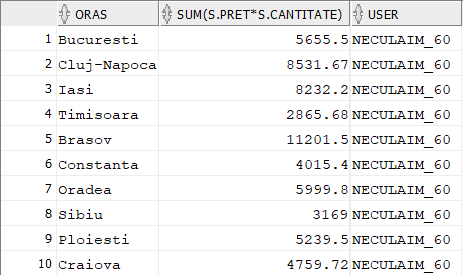
FROM locatii\_pr l, depozite d, stocuri s, produse\_pr p

WHERE p.id\_produs=s.id\_produs

AND d.id\_depozit=s.id\_depozit

AND l.id\_locatie=d.id\_locatie

GROUP BY l.oras;



1. Să se selecteze denumirea, numele si prenumele responsabilului depozitelor care au valoarea totala a stocurilor de produse mai mare de 4500 de lei.

SELECT d.id\_depozit,d.denumire\_depozit,d.nume\_responsabil ,sum(s.pret\*s.cantitate) Valoare\_Toatala

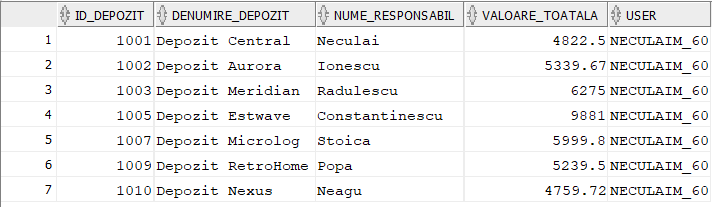
FROM produse\_pr p,stocuri s,depozite d

WHERE p.id\_produs=s.id\_produs

AND d.id\_depozit=s.id\_depozit

GROUP BY d.id\_depozit, d.denumire\_depozit, d.nume\_responsabil

HAVING sum(s.pret\*s.cantitate)>4500;



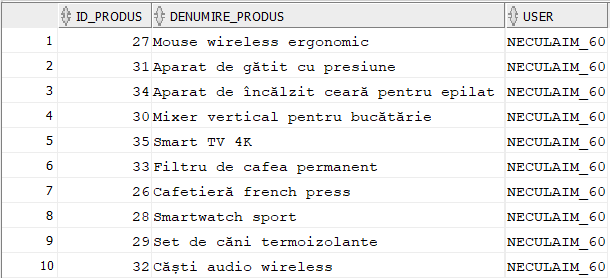
1. Să se selecteze id-ul și denumire fiecarui produs care nu se află în nici unul din stocurile depozitelor.

SELECT p.id\_produs,p.denumire\_produs

FROM produse\_pr p,stocuri s

WHERE p.id\_produs=s.id\_produs(+)

AND id\_stoc is null;

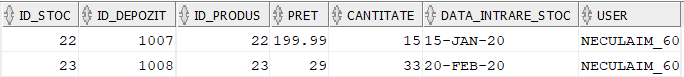


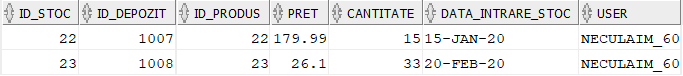
1. Să se modifice prețul produselor care sunt în stoc din anul 2020 astfel încât acesta să scadă cu 10%.

UPDATE stocuri

SET pret = pret \* 0.9

WHERE EXTRACT(YEAR FROM data\_intrare\_stoc) = 2020;

ÎNAINTE

DUPĂ

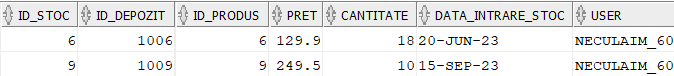
1. Să se modifice cantitate produselor care sunt în stoc și fac parte din categoria produselor de ‘GRADINĂRIT’ astfel încât aceasta să scadă cu 5 unități.

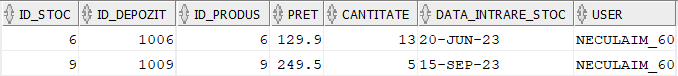
UPDATE stocuri

SET cantitate=cantitate-5

WHERE stocuri.id\_produs in (

SELECT id\_produs FROM produse\_pr WHERE UPPER(categorie\_produs) like 'GRADINĂRIT');

 ÎNAINTE

 DUPĂ

1. Să se selecteze toate produsele din depozitul care are denumirea ‘Zenith’ și a căror dată de intrare în stoc este in luna martie.

SELECT p.id\_produs,p.denumire\_produs,s.data\_intrare\_stoc

FROM stocuri s,produse\_pr p,depozite d

WHERE p.id\_produs=s.id\_produs

AND d.id\_depozit=s.id\_depozit

AND TO\_CHAR (data\_intrare\_stoc) LIKE '%MAR%'

AND LOWER(d.denumire\_depozit) LIKE '%zenith%' ;



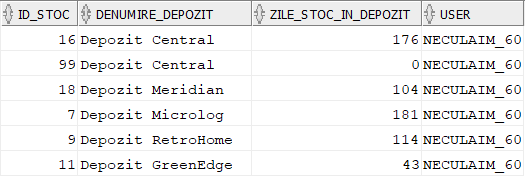
1. Să se selecteze toate stocurile care se afla in depozite de mai putin de 200 de zile si denumirea depozitului din care face parte fiecare.

SELECT s.id\_stoc,d.denumire\_depozit,TRUNC(sysdate - data\_intrare\_stoc) zile\_stoc\_in\_depozit

FROM stocuri s,depozite d

WHERE d.id\_depozit=s.id\_depozit

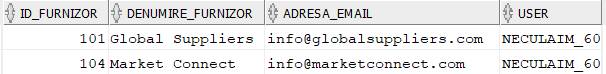
AND sysdate - data\_intrare\_stoc < 200;



1. Să se selecteze toti furnizorii a caror adresa de email incepe cu ‘info’.

SELECT \* FROM furnizori

WHERE LOWER( SUBSTR (adresa\_email, 0, 4)) ='info';



1. Să se determine diferit disponibilitatea fiecărui produs prezet în toate depozitele în funcție de cantitatea totala disponibila din acel produs astfel:
   1. Când cantitatea este mai mica de 20 de bucăți atunci disponibilitatea este ‘Stoc limitat’
   2. Când cantitatea este cuprinsă între 20 și 35 de bucăți atunci disponibilitatea este ‘Stoc moderat’
   3. Când cantitatea este mai mare de 35 de bucăți atunci disponibilitatea este ‘Stoc suficient’

Să se afiseze inregistrarile in ordine crescătoare a cantității totale.

SELECT p.id\_produs,p.denumire\_produs,sum(s.cantitate),

CASE

WHEN SUM(s.cantitate)<20 THEN 'Stoc limitat'

WHEN SUM(s.cantitate) BETWEEN 20 AND 35 THEN 'Stoc moderat'

WHEN SUM(s.cantitate)>35 THEN 'Stoc suficient'

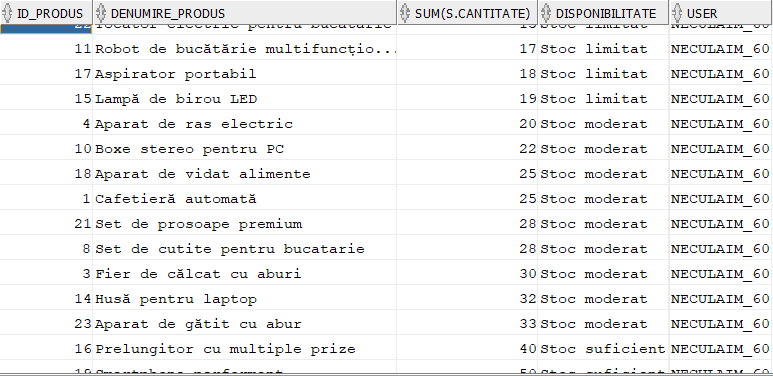
END disponibilitate

FROM produse\_pr p,stocuri s

WHERE p.id\_produs=s.id\_produs

GROUP BY p.id\_produs,p.denumire\_produs

ORDER BY SUM(s.cantitate);

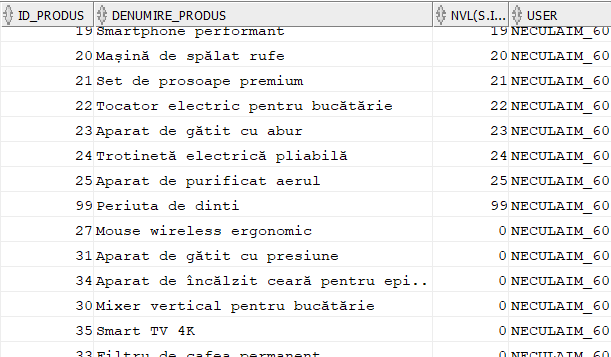


1. Să se selecteze id-ul produsului, denumirea produsului și id-ul stocului din care face parte pentru toate produsele, chiar și cele care nu se afla în nici un stoc. Pentru produsele care nu se afla în nici un stoc, valoare NULL din coloana ‘id\_stoc’ să se înlocuiască cu cifra 0.

SELECT p.id\_produs,p.denumire\_produs,NVL(s.id\_stoc,0)

FROM produse\_pr p,stocuri s

WHERE p.id\_produs=s.id\_produs(+);



1. Să se afișeze atât depozitele care au în stoc produse din categoria ‘GRADINĂRIT’ cât și depozitele care au în stoc produse din categoria ‘ACCESORII’.

SELECT d.\* FROM depozite d,stocuri s, produse\_pr p

WHERE d.id\_depozit=s.id\_depozit

AND s.id\_produs=p.id\_produs

AND UPPER(p.categorie\_produs) LIKE 'ACCESORII'

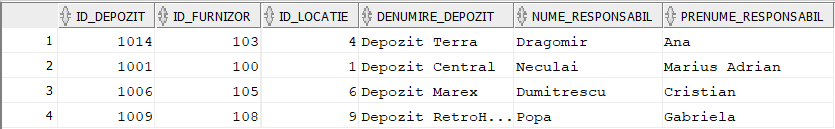
UNION

SELECT d.\* FROM depozite d,stocuri s, produse\_pr p

WHERE d.id\_depozit=s.id\_depozit

AND s.id\_produs=p.id\_produs

AND UPPER(p.categorie\_produs) LIKE 'GRADINĂRIT';



1. Sa se creeze o tabelă vitruală care sa conțină id-ul și denumire depozitelor, denumirea furnizorilor fiecărui depozit, orașul, adresa și codul poștal aferent fiecărui depozit. Înregistrările să se ordoneze după id-ul depozitelor.

CREATE VIEW gestiune\_depozite AS

SELECT d.id\_depozit ,d.denumire\_depozit,f.denumire\_furnizor,l.oras,l.adresa,l.cod\_postal

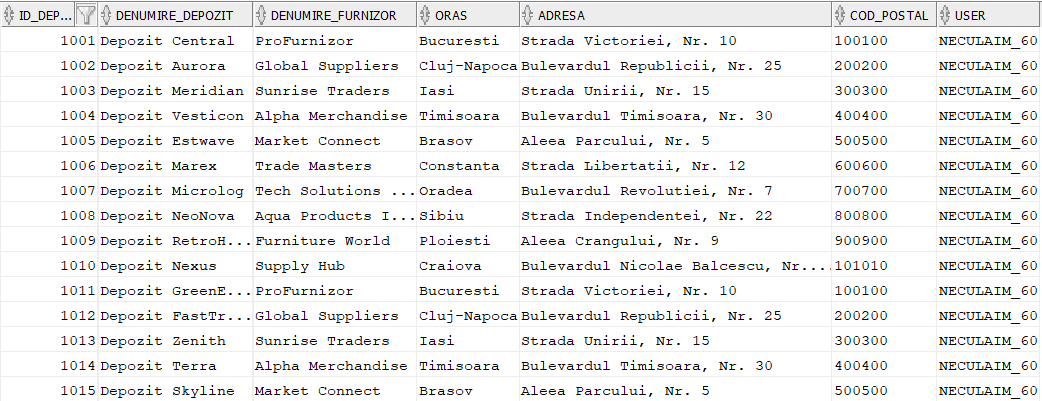
FROM depozite d,locatii\_pr l,furnizori f

WHERE d.id\_locatie=l.id\_locatie

AND d.id\_furnizor=f.id\_furnizor

ORDER BY d.id\_depozit;

SELECT \* FROM gestiune\_depozite;



1. Să se afișeze toate produsele și cantitățile disponibile în depozite mai puțin cele care fac parte din categoria ‘Bucătărie’, ‘Gradinărit’ și ’Accesorii’.

SELECT p.id\_produs,p.denumire\_produs,sum(s.cantitate)

FROM produse\_pr p,stocuri s

WHERE p.id\_produs=s.id\_produs

GROUP BY p.id\_produs,p.denumire\_produs

MINUS

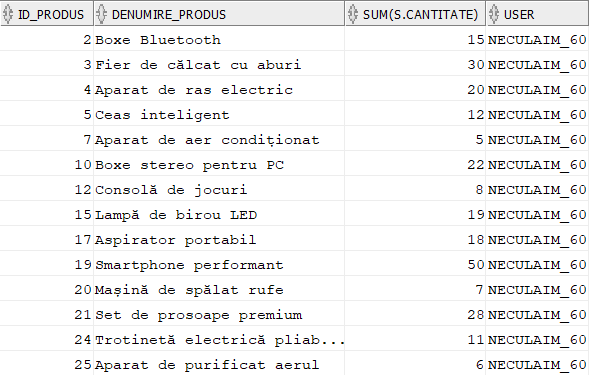
SELECT p.id\_produs,p.denumire\_produs,sum(s.cantitate)

FROM produse\_pr p,stocuri s

WHERE p.id\_produs=s.id\_produs

AND (UPPER(p.categorie\_produs)='BUCĂTĂRIE' or UPPER(p.categorie\_produs)='GRADINĂRIT' or UPPER(p.categorie\_produs)='ACCESORII')

GROUP BY p.id\_produs,p.denumire\_produs;



1. Să se introducă un nou produs în tabela produse\_pr cu id=99, denumire=’Periuta de dinti’, categorie=’Ingrijire personala’.

Să se creeze un nou stoc în tabela stocuri cu id=99, pentru depozitul cu id=1001 care să conțină 10 bucăți din produsul cu id=99 la prețul de 25 lei și data de intrare în stoc să fie data curentă a sistemului.

INSERT INTO produse\_pr (id\_produs, denumire\_produs, categorie\_produs)

VALUES (99, 'Periuta de dinti', 'Ingrijire personala');



INSERT INTO stocuri (id\_stoc, id\_depozit, id\_produs, pret, cantitate, data\_intrare\_stoc)

VALUES (99, 1001, 99, 25, 10, SYSDATE);



1. Să se selecteze id-ul produselor care au fost furnizate de furnizorul cu id\_furnizor=100, denumire\_produs și id-ul depozitului din care face parte.

SELECT DISTINCT p.id\_produs,p.denumire\_produs,d.id\_depozit

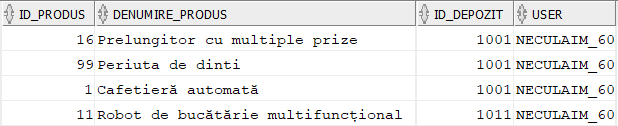
FROM produse\_pr p,furnizori f,depozite d,stocuri s

WHERE f.id\_furnizor=d.id\_furnizor

AND d.id\_depozit=s.id\_depozit

AND s.id\_produs=p.id\_produs

AND f.id\_furnizor=100;



1. Să se selecteze locațiile în care depozitele au mai mult de 40 de bucati de produse. Înregistrările să se afiseze în ordine crescătoare dupa numărul de produse.

SELECT l.\*,sum(s.cantitate) nr\_buc\_produse

FROM produse\_pr p,depozite d,stocuri s,locatii\_pr l

WHERE l.id\_locatie=d.id\_locatie

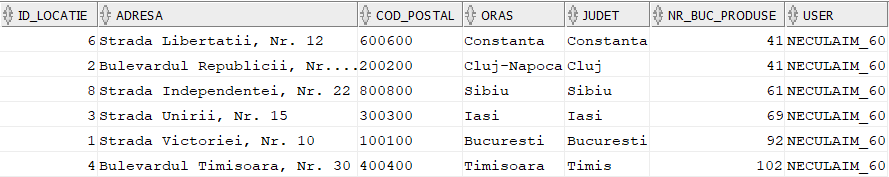
AND d.id\_depozit=s.id\_depozit

AND p.id\_produs=s.id\_produs

GROUP BY l.id\_locatie,l.adresa,l.cod\_postal,l.oras,l.judet

HAVING sum(s.cantitate)>40

ORDER BY sum(s.cantitate);



1. Să se șteargă iar apoi să se recupereze tabela ‘locatii\_pr’.

DROP TABLE stocuri cascade constraints;

FLASHBACK TABLE stocuri TO BEFORE DROP;

--refacerea legaturilor dintre tabele

ALTER TABLE stocuri

ADD CONSTRAINT fk\_stocuri\_d FOREIGN KEY(id\_depozit) REFERENCES depozite(id\_depozit);

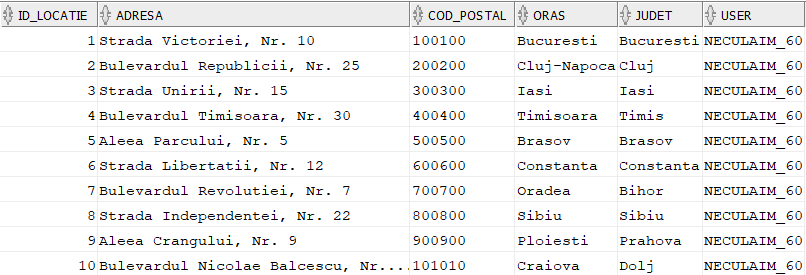
ALTER TABLE stocuri

ADD CONSTRAINT fk\_stocuri\_p FOREIGN KEY(id\_produs) REFERENCES produse\_pr(id\_produs);

1. Să se creeze un sinonim pentru tabela locatii\_pr cu denumirea ‘locatii\_depozite’.

CREATE SYNONYM locatii\_depozite FOR locatii\_pr;

SELECT \* FROM locatii\_depozite;

1. Să se creeze un raport în care să se afiseze cate produse sunt în fiecare depozit, totalul cantitatilor de produse din fiecare depozit, si totalul pentru toate depozitele.

SELECT NVL(TO\_CHAR(d.id\_depozit),'Toate depozitele') Depozit,

NVL(TO\_CHAR(s.id\_produs),'Total cantitate produse') Produs

,SUM(s.cantitate) cantitate ,LPAD(' ',5)||user as Utilizator

FROM stocuri s,depozite d

WHERE s.id\_depozit=d.id\_depozit

GROUP BY ROLLUP(d.id\_depozit,s.id\_produs)

ORDER BY d.id\_depozit;

