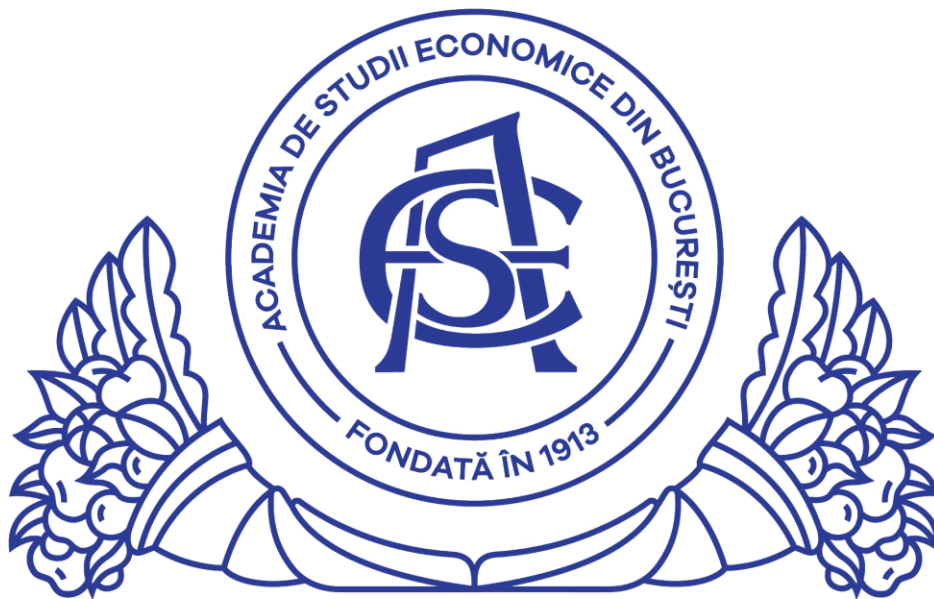


Proiect – Baze de date

Tema: Magazin Electronice/Electrocasnice



Țucă Mădălin Gabriel

Seria C

Grupa 1048

Cuprins:

- Descrierea temei.....3.
- Schema conceptuală și explicații.....4.
- Construcția bazei de date, operații LDD.....5.
- Operații de actualizare a datelor.....
- Interogări variate folosind funcțiile menționate
- Script SQL.....6

1. Descrierea temei

Baza de date a proiectului exemplifică un magazin de electronice/electrocasnice și ajută la gestionarea produselor și distribuirea acestora către clienți.

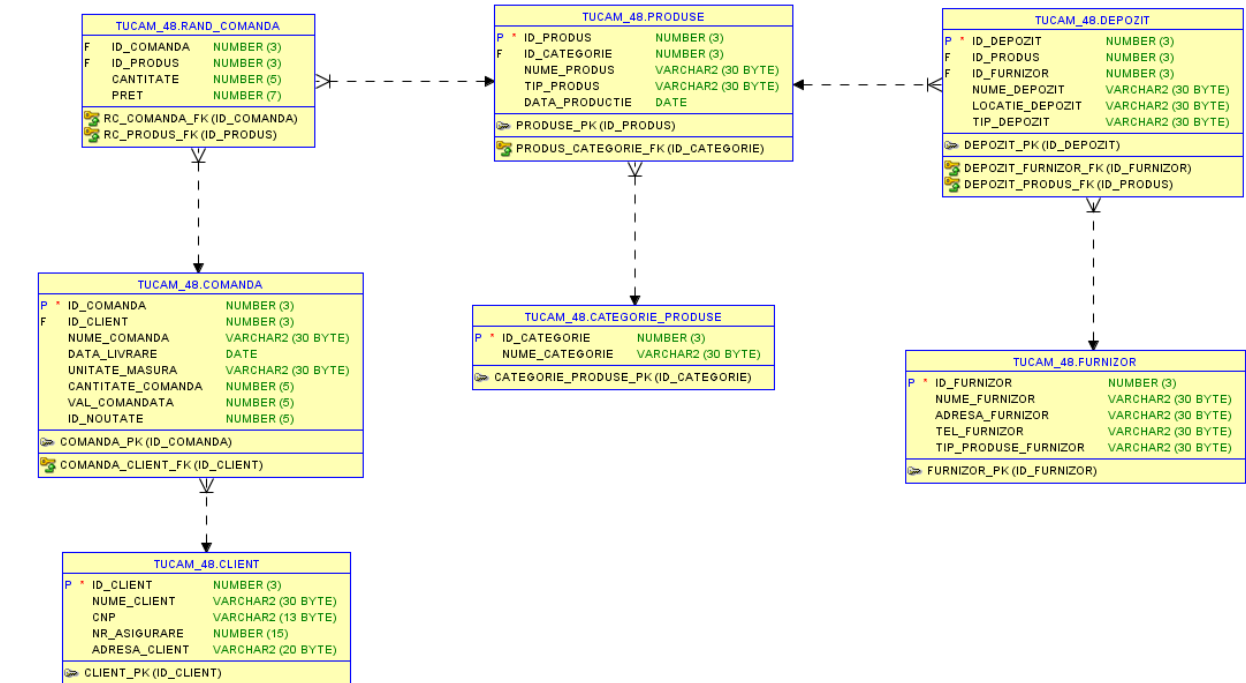
Tabelele bazei de date țin evidența produselor, categoriilor de produse, depozitelor și furnizorilor, clienților înregistrați și comenzile acestora. Operațiunile realizate pe baza de date ajută la filtrarea și identificarea anumitor cerințe de vizualizare a datelor.

Deoarece un magazin este dependent de evidența produselor sale cu scopul de a le gestiona într-un mod cât mai optim, acestea trebuie să fie stocate undeva pentru a le privi din ansamblu. Cum tehnologia avansează din ce în ce mai rapid și vine în sprijinul activităților economice, folosirea bazelor de date informatice de tip Oracle, mySQL, PostgreSQL etc. a devenit din ce în ce mai indispensabilă prin facilitarea operațiilor cu datele.

Fie un magazin fictiv de electronice/electrocasnice, acesta având în stocul său produse electronice pe diferite categorii (telefoane, PC-uri, mașini de spălat etc.). Pentru vizualizarea datelor într-un mod cât mai facil, se vor folosi diferite funcționalități pentru sortare, calculare de date numerice, afișarea depozitelor unde produsele sunt disponibile, a clienților și comenzile acestora, alături de diferite interogări ce ajută în acest sens.

Pentru construirea bazei de date, vom contura în primul rând schema bazei de date și legăturile dintre tabele.

2 Schema conceptuală a bazei de date



Baza de date aferentă acestui magazin este alcătuită din 7 tabele, acestea fiind: Produse, Categoria produselor, Depozite, Furnizori, Comenzi, Rând comenzi și Clienți.

Aceste tabele relaționează între ele prin tipul de legătură 1:M.

Ex: O categorie de produs are mai multe produse (Telefoane: Samsung, iPhone, Huawei etc.)

Baza de date creată este specifică magazinului nostru. Pentru stocarea produselor este nevoie de un stoc/depozit, furnizori pentru aceste depozite și distribuția produselor pe categoriile aferente, către clienți și comenzile lor.

Vom continua cu operații de tip LDD pentru construirea structurii bazei de date.

3. Operații LDD (CREATE, ALTER, DROP)

Operații cu CREATE (7 operații):

```
prompt =====
prompt Crearea tabelii Client
prompt =====
CREATE TABLE Client (
  id_client NUMBER(3) PRIMARY KEY,
  nume_client VARCHAR2(30),
  CNP VARCHAR2(13),
  nr_asigurare NUMBER(15),
  adresa_client VARCHAR2(20)
);

prompt =====
prompt Crearea tabelii Rand_Comanda
prompt =====
CREATE TABLE Rand_Comanda (
  id_produș NUMBER(3),
  id_comanda NUMBER(3),
  cantitate NUMBER(5),
  pret NUMBER(7)
);

prompt =====
prompt Crearea tabelii Client
prompt =====
CREATE TABLE Client (
  id_client NUMBER(3) PRIMARY KEY,
  nume_client VARCHAR2(30),
  CNP VARCHAR2(13),
  nr_asigurare NUMBER(15),
  adresa_client VARCHAR2(20)
);

prompt =====
prompt Crearea tabelii Rand_Comanda
prompt =====
CREATE TABLE Rand_Comanda (
  id_produș NUMBER(3),
  id_comanda NUMBER(3),
  cantitate NUMBER(5),
  pret NUMBER(7)
);
```

Operații cu ALTER pentru adăugare CONSTRAINTS(6 operații):

```
prompt =====
prompt Adaugare de Foreign Keys
prompt =====

ALTER TABLE Produse ADD CONSTRAINT produs_categorie_FK FOREIGN KEY(id_categorie)
REFERENCES Categorie_Produse(id_categorie);
ALTER TABLE Depozit ADD CONSTRAINT depozit_furnizor_FK FOREIGN KEY(id_furnizor)
REFERENCES Furnizor(id_furnizor);
ALTER TABLE Comanda ADD CONSTRAINT comanda_client_FK FOREIGN KEY(id_client)
REFERENCES Client(id_client);
ALTER TABLE Rand_Comanda ADD CONSTRAINT rc_produs_FK FOREIGN KEY(id_produs)
REFERENCES Produse(id_produs);
ALTER TABLE Rand_Comanda ADD CONSTRAINT rc_comanda FOREIGN KEY(id_comanda)
REFERENCES Comanda(id_comanda);
ALTER TABLE Depozit ADD CONSTRAINT depozit_produs_FK FOREIGN KEY(id_produs)
REFERENCES Produse(id_produs);
```

Operații cu DROP(7 operații):

```
prompt =====
prompt Drop optional
prompt =====

DROP TABLE CATEGORIE_PRODUSE CASCADE CONSTRAINTS;
DROP TABLE CLIENT CASCADE CONSTRAINTS;
DROP TABLE COMANDA CASCADE CONSTRAINTS;
DROP TABLE DEPOZIT CASCADE CONSTRAINTS;
DROP TABLE FURNIZOR CASCADE CONSTRAINTS;
DROP TABLE PRODUSE CASCADE CONSTRAINTS;
DROP TABLE RAND_COMANDA CASCADE CONSTRAINTS;
```

4. Operații cu INSERT pentru fiecare tabelă:

```
prompt =====
prompt Inserare valori categorie produse
prompt =====
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (1, 'Telefoane');
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (2, 'PC-uri');
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (3, 'Laptop-uri');
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (4, 'Tablete');
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (5, 'Electrocasnice');
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (6, 'SmartWatch');
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (7, 'Camere Foto');
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (8, 'Camere Video');
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (9, 'Imprimante');
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (10, 'Scannere');
SELECT * FROM CATEGORIE_PRODUSE;
```

```
prompt =====
prompt Inserare valori in produse
prompt =====
INSERT INTO PRODUSE VALUES(1,1,'Samsung', 'Galaxy S9', TO_DATE('01-20-2019','mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(2,1,'iPhone', '10', TO_DATE('05-25-2019','mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(3,1,'Huawei', 'P30 Pro', TO_DATE('05-25-2019','mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(4,2,'Alienware', 'Gaming', TO_DATE('05-25-2019','mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(5,2,'ASUS', 'xTreme', TO_DATE('03-21-2019','mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(6,3,'Lenovo', 'IdeaPad 700', TO_DATE('06-10-2017', 'mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(7,3,'macBook', 'Pro', TO_DATE('10-10-2019', 'mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(8,4,'Samsung', 'Note 4', TO_DATE('04-08-2018', 'mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(9,5,'Whirlpool', '6th sense', TO_DATE('12-02-2015', 'mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(10,6,'Samsung', 'Gear S3 Frontier', TO_DATE('05-05-2017', 'mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(11,6,'Apple', 'Series 3', TO_DATE('03-02-2018', 'mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(12,7,'Nikon', 'D850', TO_DATE('04-07-2016', 'mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(13,8,'Sony', 'FDR-AX33', TO_DATE('07-19-2019', 'mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(14,9,'HP', 'Deskjet 12', TO_DATE('02-24-2017', 'mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(15,9,'Canon', 'PIXMA E3170', TO_DATE('12-21-2016', 'mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(16,10,'Canon', 'CanoScan LiDE 400', TO_DATE('05-21-2019', 'mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(17,10,'Kodak', 'i3300', TO_DATE('04-20-2019', 'mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(18,10,'HP', 'rx321', NULL);
INSERT INTO PRODUSE VALUES(19,2,'Acer', 'Predator Orion 9000', NULL);
INSERT INTO PRODUSE VALUES(20,2,'Lenovo', 'Legion', NULL);
INSERT INTO PRODUSE VALUES(21,1,'Sony Ericsson', 'w880i', TO_DATE('02-01-2004', 'mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(22,1,'Sony Ericsson', 'p250i', TO_DATE('12-04-2002', 'mm-dd-yyyy'));
SELECT * FROM PRODUSE ORDER BY id_produș ASC;
```

```

prompt =====
prompt Inserare in tabela rand_comanda
prompt =====
DESC RAND_COMANDA;
INSERT INTO RAND_COMANDA VALUES (1,6,2,6300);
INSERT INTO RAND_COMANDA VALUES (2,1,3,3000);
INSERT INTO RAND_COMANDA VALUES (3,9,2,4000);
INSERT INTO RAND_COMANDA VALUES (4,7,1,3000);
INSERT INTO RAND_COMANDA VALUES (5,4,1,5400);
INSERT INTO RAND_COMANDA VALUES (6,14,1,1200);
INSERT INTO RAND_COMANDA VALUES (7,16,1,700);

SELECT * FROM RAND_COMANDA;

prompt =====
prompt Inserare in tabela depozit
prompt =====

INSERT INTO DEPOZIT VALUES(1,1,1,'Depozit Altex', 'Sos.Berceni', 'Intern');
INSERT INTO DEPOZIT VALUES(2,2,1,'Depozit Altex', 'Sos.Berceni', 'Intern');
INSERT INTO DEPOZIT VALUES(3,3,1,'Depozit Altex', 'Sos.Berceni', 'Intern');
INSERT INTO DEPOZIT VALUES(4,9,2,'Depozit Flanco', 'Grozavesti', 'Extern');
INSERT INTO DEPOZIT VALUES(5,12,3, 'Depozit eMAG', 'Crangasi' , 'Intern');
SELECT * FROM DEPOZIT;

prompt =====
prompt Inserare date in tabela furnizor
prompt =====
DESC FURNIZOR;
INSERT INTO FURNIZOR VALUES(1,'Altex','Militari, Bd.Iuliu Maniu', '0723319930', 'Telefoane');
INSERT INTO FURNIZOR VALUES(2,'Flanco', 'Berceni, Str.Doamna Ghica', '0761503342', 'Electrocasnice');
INSERT INTO FURNIZOR VALUES(3,'eMAG', 'Crangasi, Str. Crangasi', '0733918830', 'Camere Foto/Video');
INSERT INTO FURNIZOR VALUES(4,'PC Garage', 'Grozavesti, Str. Grozavesti', '0723954821', 'PC/Laptop');
SELECT * FROM FURNIZOR;

prompt =====
prompt Inserare date in tabela clienti
prompt =====
DESC CLIENT;
INSERT INTO CLIENT VALUES (1,'Tuca Madalin', '1990911162022', '254383', 'Str. Judetului 9');
INSERT INTO CLIENT VALUES (2,'Cezin Cupii', '1981024134291', '34950210', 'Belvedere A7');
INSERT INTO CLIENT VALUES (3,'Robert Eftene', '1990523239022', '431231', 'Belvedere A7');
INSERT INTO CLIENT VALUES (4,'Alexandru Dragomir', '198032252033', '231231', 'Str.Preciziei');
INSERT INTO CLIENT VALUES (5,'Pavel Craciun', '1990912152044', '111043', 'Camin Tei');
INSERT INTO CLIENT VALUES (6,'Romel Burcea', '1991013123456', '2303204', 'Str.Mirodeniei');
INSERT INTO CLIENT VALUES (7,'Tiberiu Lazar', '1990305122445', '343141', 'Belvedere A7');
INSERT INTO CLIENT VALUES (8,'George Dolofan', '1991122456785', '42414421', 'Timisoara, Poli');
SELECT * FROM CLIENT;

prompt =====
prompt Inserare in tabela comanda
prompt =====

DESC COMANDA;
INSERT INTO COMANDA VALUES (1,1,'Laptop-uri', TO_DATE('jan 5,2020', 'mon dd, yyyy'), 'buc', 2, 6300);
INSERT INTO COMANDA VALUES (2,1,'Telefoane', TO_DATE('jan 6,2020', 'mon dd, yyyy'), 'buc', 3, 3000);
INSERT INTO COMANDA VALUES (3,2,'Electrocasnice', TO_DATE('jan 8, 2020', 'mon dd,yyyy'), 'buc', 2, 4000);
INSERT INTO COMANDA VALUES (4,2,'Camere Foto', TO_DATE('jan 12, 2020', 'mon dd, yyyy'), 'buc', 1, 3000);
INSERT INTO COMANDA VALUES (5,3,'PC-uri', TO_DATE('feb 02, 2020', 'mon dd, yyyy'), 'buc', 1, 5400);
INSERT INTO COMANDA VALUES (6,4,'Imprimante', TO_DATE('feb 04, 2020', 'mon dd, yyyy'), 'buc', 1, 1200);
INSERT INTO COMANDA VALUES (7,5,'Scanere', TO_DATE('mar 20, 2020', 'mon dd, yyyy'), 'buc', 1, 700);
INSERT INTO COMANDA VALUES (8,8, 'SmartWatch', TO_DATE('jan 23, 2020', 'mon dd, yyyy'), 'buc', 1, 850);
SELECT * FROM COMANDA;

```


Operații cu DELETE (1 operație):

Sterge înregistrările din tabela PRODUSE

```
prompt =====
prompt Operatii cu DELETE
prompt =====
SET SERVEROUTPUT ON;
ALTER TABLE PRODUSE DROP CONSTRAINT DEPOZIT_PRODUS_FK;
DECLARE
    v_counter NUMBER;
BEGIN
    FOR v_counter IN 1..22 LOOP
        DELETE FROM PRODUSE WHERE id_produs = v_counter;
    END LOOP;
END;

SELECT * FROM PRODUSE;
```

5. Interogări Variate

Mai jos vor fi exemplificate 4 dintre cele 26 de interogări variate folosind funcțiile menționate

1. Afișarea produselor ale căror prețuri nu sunt stabilite și nu sunt comandate

```
SELECT pr.num_e_produs Nume, pr.tip_produs Tip, pr.data_productie Data_Fabricatie,
NVL(TO_CHAR(rcom.pret * rcom.cantitate), 'Pret nestabilit') Valoare, NVL((com.num_e_comanda), 'Nu este comandat')
Comanda,
com.data_livrare Livrare FROM ((pr
FULL OUTER JOIN rcom ON pr.id_produs = rcom.id_produs
FULL OUTER JOIN com ON rcom.id_comanda = com.id_comanda));
```

	NUME	TIP	DATA_FABRICATIE	VALOARE	COMANDA	LIVRARE
1	Samsung	Galaxy S9	20-JAN-19	9000	Telefoane Smart	06-JAN-20
2	iPhone	10	25-MAY-19	Pret nestabilit	Nu este comandat	(null)
3	Huawei	P30 Pro	25-MAY-19	Pret nestabilit	Nu este comandat	(null)
4	Alienware	Gaming	25-MAY-19	5400	PC-uri	02-FEB-20
5	ASUS	xTreme	21-MAR-19	Pret nestabilit	Nu este comandat	(null)
6	Lenovo	IdeaPad 700	10-JUN-17	12600	Laptop-uri	05-JAN-20
7	macBook	Pro	10-OCT-19	3000	Camere Foto	12-JAN-20
8	Samsung	Note 4	08-APR-18	Pret nestabilit	Nu este comandat	(null)
9	Whirlpool	6th sense	02-DEC-15	8000	Electrocasnice	08-JAN-20

2. Afișarea comenzilor ale căror valori sunt mai mari decât valoarea comenzii cu id-ul 1

```
prompt Utilizare CASE
SELECT p.id_produș, p.numē_produș, rcom.pret,
CASE WHEN rcom.pret > 3000 THEN 'Produs de ultima generație'
ELSE 'Produs de buget' END
FROM produse p, rand_comanda rcom WHERE p.id_produș = rcom.id_produș;
```

ript Output x | Query Result x | Query Result 1 x

SQL | All Rows Fetched: 7 in 0.038 seconds

ID_PRODUS	NUME_PRODUS	PRET	CASEWHENR.COM.PRET>3000THEN'PRODUSDEULTIMAGENERATIE'ELSE'PRODUSDEBUGET'END
1	1 Samsung	3000	Produs de buget
2	4 Alienware	5400	Produs de ultima generație
3	6 Lenovo	6300	Produs de ultima generație
4	7 macBook	3000	Produs de buget
5	9 Whirlpool	4000	Produs de ultima generație
6	14 HP	1200	Produs de buget
7	16 Canon	700	Produs de buget

3. Să se afișeze un mesaj de tip “Produs de ultimă generație” dacă prețul este mai mare decât 3000 de lei și un mesaj de tip “Produs de buget” pentru produsele sub 3000 de lei

```
prompt Utilizare CASE
SELECT p.id_produș, p.numē_produș, rcom.pret,
CASE WHEN rcom.pret > 3000 THEN 'Produs de ultima generație'
ELSE 'Produs de buget' END
FROM produse p, rand_comanda rcom WHERE p.id_produș = rcom.id_produș;
```

ript Output x | Query Result x | Query Result 1 x

SQL | All Rows Fetched: 7 in 0.038 seconds

ID_PRODUS	NUME_PRODUS	PRET	CASEWHENR.COM.PRET>3000THEN'PRODUSDEULTIMAGENERATIE'ELSE'PRODUSDEBUGET'END
1	1 Samsung	3000	Produs de buget
2	4 Alienware	5400	Produs de ultima generație
3	6 Lenovo	6300	Produs de ultima generație
4	7 macBook	3000	Produs de buget
5	9 Whirlpool	4000	Produs de ultima generație
6	14 HP	1200	Produs de buget
7	16 Canon	700	Produs de buget

4. Afișarea depozitelor situate în cartierul Berceni

```
prompt selectare id-ului, numelui si locatia depozitelor din Berceni
SELECT id_depozit ID, nume_depozit Depozit, locatie_depozit Locatie
FROM Depozit
WHERE SUBSTR(locatie_depozit,2,10) LIKE '%Berceni';
```

ript Output x | Query Result x | Query Result 1 x | Query Result 2 x | Query Result 3

SQL | All Rows Fetched: 3 in 0.038 seconds

ID	DEPOZIT	LOCATIE
1	1 Depozit Altex	Sos.Berceni
2	2 Depozit Altex	Sos.Berceni
3	3 Depozit Altex	Sos.Berceni

Acestea sunt câteva din interogările efectuate, iar mai jos voi atașa tot scriptul SQL:

```
CREATE TABLE Depozit (
id_depozit NUMBER(3) PRIMARY KEY,
id_produs NUMBER(3),
id_furnizor NUMBER(3),
nume_depozit VARCHAR2(30),
locatie_depozit VARCHAR2(30),
tip_depozit VARCHAR2(30)
);
```

prompt =====

prompt Crearea tabelii Furnizor

prompt =====

```
CREATE TABLE Furnizor (
id_furnizor NUMBER(3) PRIMARY KEY,
nume_furnizor VARCHAR2(30),
```

```
adresa_furnizor VARCHAR2(30),  
telefon_furnizor VARCHAR2(30),  
tip_produce_furnizor VARCHAR2(30)  
);
```

```
prompt =====  
prompt Crearea tabeli Produs  
prompt =====
```

```
CREATE TABLE Produse (  
id_produs NUMBER(3) PRIMARY KEY,  
id_categorie NUMBER(3),  
nume_produs VARCHAR2(30),  
tip_produs VARCHAR2(30),  
data_productie DATE  
);
```

```
prompt =====  
prompt Crearea tabeli Categorie_Produs  
prompt =====
```

```
CREATE TABLE Categorie_Produse (  
id_categorie NUMBER(3) PRIMARY KEY,  
nume_categorie VARCHAR2(30)  
);
```

prompt =====

prompt Crearea tabelii Comanda

prompt =====

```
CREATE TABLE Comanda (  
id_comanda NUMBER(3) PRIMARY KEY,  
id_client NUMBER(3),  
nume_comanda VARCHAR2(30),  
data_livrare DATE,  
unitate_masura VARCHAR2(5),  
cantitate_comandata NUMBER(5),  
val_comandata NUMBER(5)  
);
```

prompt =====

prompt Crearea tabelii Client

prompt =====

```
CREATE TABLE Client (  
id_client NUMBER(3) PRIMARY KEY,  
nume_client VARCHAR2(30),  
CNP VARCHAR2(13),  
nr_asigurare NUMBER(15),  
adresa_client VARCHAR2(20));
```

prompt =====

prompt Crearea tabelii Rand_Comanda

prompt =====

```
CREATE TABLE Rand_Comanda (  
id_produs NUMBER(3),  
id_comanda NUMBER(3),  
cantitate NUMBER(5),  
pret NUMBER(7));
```

prompt =====

prompt Adaugare de Foreign Keys

prompt =====

```
ALTER TABLE Produse ADD CONSTRAINT produs_categorie_FK FOREIGN  
KEY(id_categorie)  
REFERENCES Categorie_Produse(id_categorie);  
ALTER TABLE Depozit ADD CONSTRAINT depozit_furnizor_FK FOREIGN KEY(id_furnizor)  
REFERENCES Furnizor(id_furnizor);  
ALTER TABLE Comanda ADD CONSTRAINT comanda_client_FK FOREIGN KEY(id_client)  
REFERENCES Client(id_client);  
ALTER TABLE Rand_Comanda ADD CONSTRAINT rc_produs_FK FOREIGN KEY(id_produs)  
REFERENCES Produs(id_produs);  
ALTER TABLE Rand_Comanda ADD CONSTRAINT rc_comanda FOREIGN KEY(id_comanda)  
REFERENCES Comanda(id_comanda);
```

```
ALTER TABLE Depozit ADD CONSTRAINT depozit_produs_FK FOREIGN KEY(id_produs)
REFERENCES Produs(id_produs);
```

prompt =====

prompt Drop optional

prompt =====

```
DROP TABLE CATEGORIE_PRODUSE CASCADE CONSTRAINTS;
```

```
DROP TABLE CLIENT CASCADE CONSTRAINTS;
```

```
DROP TABLE COMANDA CASCADE CONSTRAINTS;
```

```
DROP TABLE DEPOZIT CASCADE CONSTRAINTS;
```

```
DROP TABLE FURNIZOR CASCADE CONSTRATS;
```

```
DROP TABLE PRODUSE CASCADE CONSTRAINTS;
```

```
DROP TABLE RAND_COMANDA CASCADE CONSTRAINT
```

prompt =====

prompt Inserare valori categorii produse

prompt =====

```
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (1,
'Telefoane');
```

```
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (2,
'PC-uri');
```

```
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (3,
'Laptop-uri');
```

```
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (4, 'Tablete');
```

```
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (5, 'Electrocasnice');
```

```
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (6, 'SmartWatch');
```

```
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (7, 'Camere Foto');
```

```
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (8, 'Camere Video');
```

```
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (9, 'Imprimante');
```

```
INSERT INTO CATEGORIE_PRODUSE (ID_CATEGORIE, NUME_CATEGORIE) VALUES (10, 'Scannere');
```

prompt =====

prompt Inserare valori in produse

prompt =====

```
INSERT INTO PRODUSE VALUES(1,1,'Samsung', 'Galaxy S9', TO_DATE('01-20-2019','mm-dd-yyyy'));
```

```
INSERT INTO PRODUSE VALUES(2,1,'iPhone', '10', TO_DATE('05-25-2019','mm-dd-yyyy'));
```

```
INSERT INTO PRODUSE VALUES(3,1,'Huawei', 'P30 Pro', TO_DATE('05-25-2019','mm-dd-yyyy'));
```

```
INSERT INTO PRODUSE VALUES(4,2,'Alienware', 'Gaming', TO_DATE('05-25-2019','mm-dd-yyyy'));
```

```
INSERT INTO PRODUSE VALUES(5,2,'ASUS', 'xTreme', TO_DATE('03-21-2019','mm-dd-yyyy'));
```

```
INSERT INTO PRODUSE VALUES(6,3,'Lenovo', 'IdeaPad 700', TO_DATE('06-10-2017','mm-dd-yyyy'));
```



```

INSERT INTO PRODUSE VALUES(7,3,'macBook', 'Pro', TO_DATE('10-10-2019', 'mm-dd-
yyyy'));
INSERT INTO PRODUSE VALUES(8,4,'Samsung', 'Note 4', TO_DATE('04-08-2018', 'mm-dd-
yyyy'));
INSERT INTO PRODUSE VALUES(9,5,'Whirlpool', '6th sense', TO_DATE('12-02-2015', 'mm-
dd-yyyy'));
INSERT INTO PRODUSE VALUES(10,6,'Samsung', 'Gear S3 Frontier', TO_DATE('05-05-
2017', 'mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(11,6,'Apple', 'Series 3', TO_DATE('03-02-2018', 'mm-dd-
yyyy'));
INSERT INTO PRODUSE VALUES(12,7,'Nikon', 'D850', TO_DATE('04-07-2016', 'mm-dd-
yyyy'));
INSERT INTO PRODUSE VALUES(13,8,'Sony', 'FDR-AX33', TO_DATE('07-19-2019', 'mm-dd-
yyyy'));
INSERT INTO PRODUSE VALUES(14,9,'HP', 'Deskjet 12', TO_DATE('02-24-2017', 'mm-dd-
yyyy'));
INSERT INTO PRODUSE VALUES(15,9,'Canon', 'PIXMA E3170', TO_DATE('12-21-2016', 'mm-
dd-yyyy'));
INSERT INTO PRODUSE VALUES(16,10,'Canon', 'CanoScan LiDE 400', TO_DATE('05-21-
2019', 'mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(17,10,'Kodak', 'i3300', TO_DATE('04-20-2019', 'mm-dd-
yyyy'));
INSERT INTO PRODUSE VALUES(18,10,'HP', 'rx321', NULL);
INSERT INTO PRODUSE VALUES(19,2,'Acer', 'Predator Orion 9000', NULL);
INSERT INTO PRODUSE VALUES(20,2,'Lenovo', 'Legion', NULL);
INSERT INTO PRODUSE VALUES(21,1,'Sony Ericsson', 'w880i', TO_DATE('02-01-2004',
'mm-dd-yyyy'));
INSERT INTO PRODUSE VALUES(22,1,'Sony Ericsson', 'p250i', TO_DATE('12-04-2002', 'mm-
dd-yyyy'));
SELECT * FROM PRODUSE ORDER BY id_produs ASC;

```

prompt =====

prompt Inserare date in tabela furnizor

prompt =====

DESC FURNIZOR;

INSERT INTO FURNIZOR VALUES(1,'Altex','Militari, Bd.Iuliu Maniu', '0723319930',
'Telefoane');

INSERT INTO FURNIZOR VALUES(2,'Flanco', 'Berceni, Str.Doamna Ghica', '0761503342',
'Electrocasnice');

INSERT INTO FURNIZOR VALUES(3,'eMAG', 'Crangasi, Str. Crangasi', '0733918830',
'Camere Foto/Video');

INSERT INTO FURNIZOR VALUES(4,'PC Garage', 'Grozavesti, Str. Grozavesti',
'0723954821', 'PC/Laptop');

SELECT * FROM FURNIZOR;

prompt =====

prompt Inserare date in tabela clienti

prompt =====

DESC CLIENT;

INSERT INTO CLIENT VALUES (1,'Tuca Madalin', '1990911162022', '254383', 'Str. Judetului 9');

INSERT INTO CLIENT VALUES (2,'Cezin Cupii', '1981024134291', '34950210', 'Belvedere A7');

INSERT INTO CLIENT VALUES (3,'Robert Eftene', '1990523239022', '431231', 'Belvedere A7');

INSERT INTO CLIENT VALUES (4,'Alexandru Dragomir', '198032252033', '231231',
'Str.Preciziei');

```
INSERT INTO CLIENT VALUES (5,'Pavel Craciun', '1990912152044', '111043', 'Camin Tei');
```

```
INSERT INTO CLIENT VALUES (6,'Romel Burcea', '1991013123456', '2303204',  
'Str.Mirodeniei');
```

```
INSERT INTO CLIENT VALUES (7,'Tiberiu Lazar', '1990305122445', '343141', 'Belvedere A7');
```

```
INSERT INTO CLIENT VALUES (8,'George Dolofan', '1991122456785', '42414421', 'Timisoara,  
Poli');
```

```
SELECT * FROM CLIENT;
```

prompt =====

prompt Inserare in tabela comanda

prompt =====

```
DESC COMANDA;
```

```
INSERT INTO COMANDA VALUES (1,1,'Laptop-uri', TO_DATE('jan 5,2020', 'mon dd, yyyy'),  
'buc', 2, 6300);
```

```
INSERT INTO COMANDA VALUES (2,1,'Telefoane', TO_DATE('jan 6,2020', 'mon dd, yyyy'),  
'buc', 3, 3000);
```

```
INSERT INTO COMANDA VALUES (3,2,'Electrocasnice', TO_DATE('jan 8, 2020', 'mon  
dd,yyyy'), 'buc', 2, 4000);
```

```
INSERT INTO COMANDA VALUES (4,2,'Camere Foto', TO_DATE('jan 12, 2020', 'mon dd,  
yyyy'), 'buc', 1, 3000);
```

```
INSERT INTO COMANDA VALUES (5,3,'PC-uri', TO_DATE('feb 02, 2020', 'mon dd, yyyy'),  
'buc', 1, 5400);
```

```
INSERT INTO COMANDA VALUES (6,4,'Imprimante', TO_DATE('feb 04, 2020', 'mon dd, yyyy'),  
'buc', 1, 1200);
```

```
INSERT INTO COMANDA VALUES (7,5,'Scanere', TO_DATE('mar 20, 2020', 'mon dd, yyyy'),  
'buc', 1, 700);
```

```
INSERT INTO COMANDA VALUES (8,8,'SmartWatch', TO_DATE('jan 23, 2020', 'mon dd,  
yyyy'), 'buc', 1, 850);
```

```
SELECT * FROM COMANDA;
```

```
prompt =====
```

```
prompt Inserare in tabela rand_comanda
```

```
prompt =====
```

```
DESC RAND_COMANDA;
```

```
INSERT INTO RAND_COMANDA VALUES (1,6,2,6300);
```

```
INSERT INTO RAND_COMANDA VALUES (2,1,3,3000);
```

```
INSERT INTO RAND_COMANDA VALUES (3,9,2,4000);
```

```
INSERT INTO RAND_COMANDA VALUES (4,7,1,3000);
```

```
INSERT INTO RAND_COMANDA VALUES (5,4,1,5400);
```

```
INSERT INTO RAND_COMANDA VALUES (6,14,1,1200);
```

```
INSERT INTO RAND_COMANDA VALUES (7,16,1,700);
```

```
SELECT * FROM RAND_COMANDA;
```

```
prompt =====
```

```
prompt Inserare in tabela depozit
```

```
prompt =====
```

```
INSERT INTO DEPOZIT VALUES(1,1,1,'Depozit Altex', 'Sos.Berceni', 'Intern');
```

```
INSERT INTO DEPOZIT VALUES(2,2,1,'Depozit Altex', 'Sos.Berceni', 'Intern');
```

```
INSERT INTO DEPOZIT VALUES(3,3,1,'Depozit Altex', 'Sos.Berceni', 'Intern');
```

```
INSERT INTO DEPOZIT VALUES(4,9,2,'Depozit Flanco', 'Grozavesti', 'Extern');
```

```
INSERT INTO DEPOZIT VALUES(5,12,3,'Depozit eMAG', 'Crangasi', 'Intern');
```

```
SELECT * FROM DEPOZIT;
```

```
prompt =====
```

```
prompt Operatie cu DELETE
```

```
prompt =====
```

```
SET SERVEROUTPUT ON;
```

```
ALTER TABLE PRODUSE DROP CONSTRAINT DEPOZIT_PRODUS_FK;
```

```
DECLARE
```

```
    v_counter NUMBER;
```

```
BEGIN
```

```
    FOR v_counter IN 1..22 LOOP
```

```
        DELETE FROM PRODUSE WHERE id_produs = v_counter;
```

```
    END LOOP;
```

```
END;
```

```
SELECT * FROM PRODUSE;
```

```
prompt =====
```

```
prompt Interogari variate
```

```
prompt =====
```

```
prompt 1. Sa se afiseze toate produsele in functie de categoriile acestora
```

```
SELECT p.nume_produs Produs, p.tip_produs Tip, c.nume_categorie Categorie FROM  
produse p, categorie_produse c
```

```
WHERE p.id_categorie = c.id_categorie;
```

prompt 2. Afisare produse,tip,categorie,cantitate si pret total daca acesta este mai mare decat 1000 lei

```
SELECT p.nume_produș Produse, p.tip_produș Tip, c.nume_categorie Categorie,  
rc.cantitate Cantitate, (rc.pret*rc.cantitate) Val_Totala  
FROM produse p, categoriē_produse c, rand_comanda rc  
WHERE p.id_categorie = c.id_categorie AND p.id_produș = rc.id_produș AND rc.pret > 1000;
```

prompt 3. Afisare produs fara data de fabricatie

```
SELECT p.nume_produș Produse, p.tip_produș Tip, p.data_productie Data_Fabricatie  
FROM produse p WHERE p.data_productie IS NULL;
```

prompt 4. Afisare produse dupa buget introdus de la tastatura

```
accept x number prompt 'Introduceti buget maxim: '  
SELECT p.nume_produș Produse, p.tip_produș Tip, c.nume_categorie Categorie, rc.pret  
Valoare  
FROM produse p, categoriē_produse c, rand_comanda rc  
WHERE p.id_categorie = c.id_categorie AND p.id_produș = rc.id_produș AND rc.pret < &x;
```

prompt 5. Afisare produse de la marca HP folosind LIKE

```
SELECT p.nume_produș Produse, p.tip_produș Tip, c.nume_categorie Categorie  
FROM produse p, categoriē_produse c  
WHERE p.id_categorie = c.id_categorie AND p.nume_produș LIKE 'HP';
```

prompt 6. Afisare produse de la marcile Lenovo, HP, Samsung

```
SELECT * FROM produse WHERE nume_produc IN ('Lenovo', 'HP', 'Samsung');
```

prompt Afisare produse ale caror preturi se situeaza intre 1000 si 5000 de lei

```
SELECT p.nume_produc Produse, p.tip_produc Tip, rc.pret Pret  
FROM produse p, rand_comanda rc WHERE rc.pret BETWEEN 1000 AND 5000;
```

prompt Afisarea produselor in stocul furnizorilor folosind INNER JOIN

```
SELECT PRODUSE.*, DEPOZIT.NUME_DEPOZIT FROM PRODUSE INNER JOIN  
DEPOZIT ON PRODUSE.ID_PRODUS = DEPOZIT.ID_PRODUS;
```

prompt outer join pentru afisarea produselor in stoc si stoc furnizor

```
SELECT PRODUSE.*, DEPOZIT.NUME_DEPOZIT FROM PRODUSE  
LEFT OUTER JOIN DEPOZIT ON PRODUSE.ID_PRODUS = DEPOZIT.ID_PRODUS;
```

prompt afisare produse crescator dupa data fabricatie

```
SELECT PRODUSE.NUME_PRODUS, PRODUSE.TIP_PRODUS,  
PRODUSE.DATA_PRODUCTIE FROM produse ORDER BY (data_productie) ASC;
```

prompt afisarea comenzilor a caror valoare este mai mare decat valoarea comenzii cu id-ul 1

```
SELECT PRODUSE.NUME_PRODUS Produse, produse.tip_produs Tip,  
SUM(rand_comanda.pret * rand_comanda.cantitate) Suma  
FROM produse  
FULL OUTER JOIN rand_comanda  
ON produse.id_produs = rand_comanda.id_produs GROUP BY PRODUSE.NUME_PRODUS,  
PRODUSE.TIP_PRODUS  
HAVING SUM(RAND_COMANDA.PRET * RAND_COMANDA.CANTITATE) >= (SELECT  
VAL_COMANDATA FROM COMANDA WHERE ID_COMANDA = 1);
```

prompt Afisare data de fabricatie folosind EXTRACT

```
SELECT * FROM PRODUSE;  
  
SELECT id_produs, nume_produs, tip_produs, data_productie FROM produse WHERE  
EXTRACT (YEAR from data_productie)=2019 AND EXTRACT (MONTH from  
data_productie)=3 AND EXTRACT (DAY from data_productie)=21;
```

prompt Utilizare CASE

```
SELECT p.id_produs, p.nume_produs, rcom.pret,  
CASE WHEN rcom.pret > 3000 THEN 'Produs de ultima generatie'  
ELSE 'Produs de buget' END  
FROM produse p, rand_comanda rcom WHERE p.id_produs = rcom.id_produs;
```

prompt Creere sinonim pentru rand_comanda

```
CREATE SYNONYM rcom FOR rand_comanda;  
CREATE SYNONYM pr FOR produse;
```


CREATE SYNONYM com FOR comanda;

select * from rcom;

prompt Utilizare functie UNION

SELECT produse.id_produc FROM PRODUSE UNION SELECT comanda.id_comanda FROM comanda;

prompt Folosire NVL pentru inlocuirea valorilor nule cu un mesaj care nu au fost comandate si nu au valoare de comanda

```
SELECT pr.nume_produc Nume, pr.tip_produc Tip, pr.data_productie Data_Fabricatie,  
NVL(TO_CHAR(rcom.pret * rcom.cantitate), 'Pret nestabilit') Valoare,  
NVL((com.nume_comanda), 'Nu este comandat')  
Comanda,  
com.data_livrare Livrare FROM ((pr  
FULL OUTER JOIN rcom ON pr.id_produc = rcom.id_produc  
FULL OUTER JOIN com ON rcom.id_comanda = com.id_comanda));
```

prompt Folosire Decode pentru reducere pentru toate comenzile cu comanda id = 5

```
SELECT id_comanda ID, nume_comanda comanda, val_comandata valoare,  
DECODE(UPPER(id_comanda),1,2,0.1,3,0.4,4,0.4) Reducere  
FROM Comanda WHERE id_comanda < 5;
```

prompt selectare id-ului, numelui si locatia depozitelor din Berceni

```
SELECT id_depozit ID, nume_depozit Depozit, locatie_depozit Locatie  
FROM Depozit  
WHERE SUBSTR(locatie_depozit,2,10) LIKE '%Berceni';  
SELECT * FROM DEPOZIT;
```

prompt secvente pentru produs si categorie si insereare valori

```
CREATE SEQUENCE secv_categorie  
START WITH 100 INCREMENT BY 1  
MAXVALUE 500 NOCYCLE;
```

```
CREATE SEQUENCE secv_produs  
START WITH 100 INCREMENT BY 1  
MAXVALUE 500 NOCYCLE;
```

```
INSERT INTO produse VALUES(secv_produs.NEXTVAL, secv_categorie.NEXTVAL, 'Beko',  
'Electrocasnice', '10-DEC-2019');
```

```
INSERT INTO produse VALUES(secv_produs.NEXTVAL, secv_categorie.NEXTVAL, 'Bosch',  
'Electrocasnice', SYSDATE);
```

prompt Update + Select pentru produsul dezafectat din 2004

```
Update Produse SET nume_produs = 'Telefon scos din productie' WHERE id_produs IN  
(SELECT id_produs FROM produse WHERE EXTRACT (YEAR FROM data_productie) =  
2004);  
SELECT * FROM PRODUSE;
```

prompt creare tabela virtuala

```
CREATE VIEW view_client AS SELECT id_client, nume_client, adresa_client, cnp FROM client;
```

```
SELECT * FROM view_client;
```

prompt afisare clienti din tabela virtuala al caror an de nastere este 1998

```
SELECT nume_client, CNP FROM view_client WHERE CNP LIKE '%98%';
```

prompt Afisare comenzi dintr-o anumita perioada pana la o anumita perioada exceptand un anumit interval de zile

```
SELECT * FROM comanda;
```

```
SELECT * FROM comanda WHERE data_livrare BETWEEN '05-JAN-2020' AND '12-JAN-2020'
```

```
MINUS SELECT * FROM comanda WHERE data_livrare BETWEEN '06-JAN-2020' AND '08-JAN-2020';
```

```
SELECT * FROM COMANDA;
```

prompt UPDATE + SELECT

```
Update comanda SET nume_comanda = 'Telefoane Smart' WHERE id_comanda IN (SELECT id_comanda FROM comanda WHERE nume_comanda = 'Telefoane');
```

```
SELECT * FROM PRODUSE;
```

```
ALTER TABLE comanda ADD id_noutate NUMBER(5);
```

```
SELECT id_comanda, nume_comanda, id_noutate
```

```
FROM comanda
```

```
START WITH id_comanda = 1 CONNECT BY PRIOR id_comanda = id_noutate
```

```
ORDER BY id_noutate ASC;
```

```
commit;
```