Sisteme de Gestiune a Bazelor de Date
ANUL II, SERIA 24

GESTIUNEA UNUI LANT DE MAGAZINE

MEREALBE BRIANA

GRUPA 243

1. Prezentați pe scurt baza de date (utilitatea ei).

Baza de date contine informatii cu privire la un lant de magazine, vanzarile fiecarui magazin la o data anume, angajatii, salariile incasate de angajati in 2022, produsele disponibile in magazine, stocurile care au fost livrate, comenzile plasate online de catre clienti si reducerile disponibile.

Scopul acestei baze de date este de a putea tine evidenta tuturor entitatilor intr-un mod eficient si usor de accesat.

Fiecare magazin va avea un numar de angajati printre care si curieri. Curierii vor avea masini atribuite pentru a realiza transportul comenzilor.

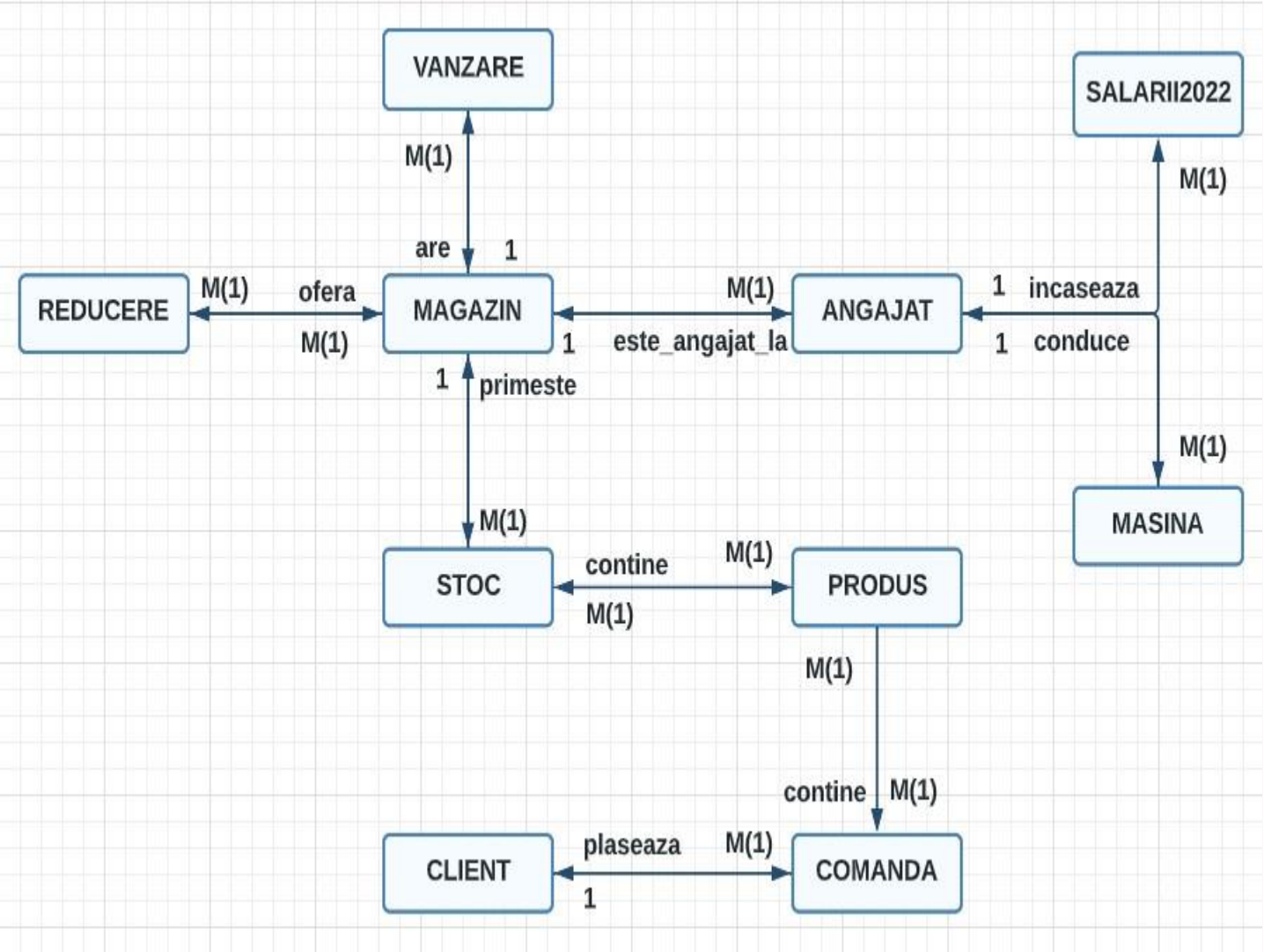
La finalul fiecarei zile lucratoare, profitul total se va trece in tabelul VANZARE pentru fiecare magazin.

Stocurile de produs sunt livrate de la firma mare catre magazinele mai mici distribuite in locatii diferite. (aceasta baza de date nu va tine evidenta si companiei mari)

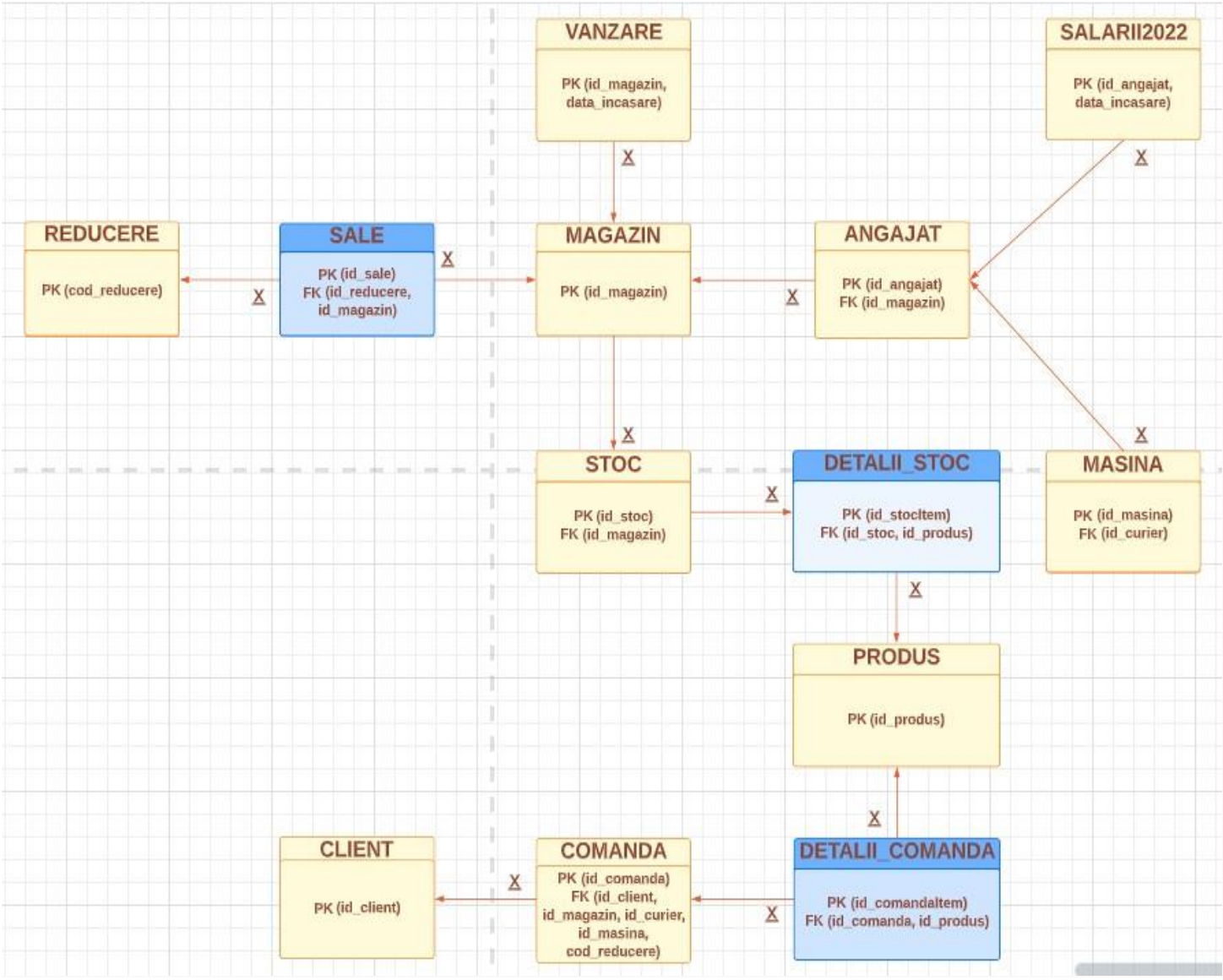
Comenzile pot fi plasate doar online de catre clientii logati la site-ul firmei.

Reducerile se aplica automat pentru comenzile online, iar pentru achizitiile facute de clientii din magazine, reducerile se aplica prin introducerea codului corespunzator in momentul in care se face plata la casa de marcat, de catre un angajat. Pentru a vedea ce reduceri sunt valabile ne uitam in tabelul SALE.

2. Realizați diagrama entitate-relație (ERD).



3. Pornind de la diagrama ERD realizați diagrama conceptuală a modelului propus, integrând toate atributele necesare.



4. Implementați în Oracle diagrama conceptuală realizată.
5. Adăugați informații coerente în tabelele create.

----------TABEL MAGAZIN----------

```
CREATE TABLE MAGAZIN (id_magazin  NUMBER(5) CONSTRAINT PKEY_MAGAZIN PRIMARY KEY,
                tara_magazin VARCHAR(500) CONSTRAINT tara_magazin NOT NULL,
                adresa_magazin VARCHAR(300) CONSTRAINT adresa_magazin NOT NULL,
                telefon_magazin VARCHAR(50) CONSTRAINT telefon_magazin NOT NULL,
                mail_magazin VARCHAR(50) CONSTRAINT mail_magazin UNIQUE NOT NULL,
                cod_postal VARCHAR(50)
                );
```

```
INSERT INTO MAGAZIN VALUES
(1,'Japonia',' 219-1130, Ikanikeisaiganaibaai, Musashino-shi,Tokyo','+8141-984-8883','TmangaSTORE-JAPAN@gmail.com','105-0013');
INSERT INTO MAGAZIN VALUES
(2,'Romania','Std Profesor Minculescu nr 34, Iasi','+40754-876-123','TmangaSTORE-ROMANIA@gmail.com','36649');
INSERT INTO MAGAZIN VALUES
(3,'Italia','Strada Provinciale 65 49, Carvico, Bergamo','+390373 5756478','TmangaSTORE-ITALY@gmail.com','54024');
INSERT INTO MAGAZIN VALUES
(4,'Germania','Oeder Weg 50, Hesse, Frankfurt','+4906728 14 75 86','TmangaSTORE-GERMANY@gmail.com','55437');
INSERT INTO MAGAZIN VALUES
(5,'UK','74 Exning Road, London',' +44070 5508 2035','TmangaSTORE-UK@gmail.com','LL46 9RT');
```

```
SELECT * FROM magazin;
```

	ID_MAGAZIN	TARA_MAGAZIN	ADRESA_MAGAZIN	TELEFON_M
1	1	Japonia	219-1130, Ikanikeisaiganaibaai, Musashino-shi,Tokyo	+8141-984-
2	2	Romania	Std Profesor Minculescu nr 34, Iasi	+40754-876
3	3	Italia	Strada Provinciale 65 49, Carvico, Bergamo	+390373 57
4	4	Germania	Oeder Weg 50, Hesse, Frankfurt	+4906728 1
5	5	UK	74 Exning Road, London	+44070 55

Sisteme de Gestiune a Bazelor de Date
ANUL II, SERIA 24

----------TABEL ANGAJAT----------

```
CREATE TABLE ANGAJAT (id_angajat NUMBER(5) CONSTRAINT PKEY_ANGAJAT PRIMARY KEY,
                      id_magazin NUMBER(5),
                      CONSTRAINT fk_angajat FOREIGN KEY(id_magazin) REFERENCES MAGAZIN (id_magazin),
                      nume_angajat VARCHAR(50) CONSTRAINT nume_angajat NOT NULL,
                      prenume_angajat VARCHAR(50) CONSTRAINT prenume_angajat NOT NULL,
                      varsta_angajat INT CONSTRAINT varsta_angajat NOT NULL,
                      sex_angajat VARCHAR(50) CONSTRAINT sex_angajat NOT NULL,
                      telefon_angajat VARCHAR(50) CONSTRAINT telefon_angajat NOT NULL,
                      functie VARCHAR(50) CONSTRAINT functie NOT NULL,
                      data_angajare DATE CONSTRAINT data_angajare NOT NULL
                      );


INSERT INTO ANGAJAT VALUES
(110,1,'Domen','Tsukiko',23,'barbat','+8199-159-0988','manager',TO_DATE('20211209', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(111,1,'Yogi','Toshio',42,'barbat','+8192-099-0119','curier',TO_DATE('20220103', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(112,1,'Arima','Yasuo',27,'femeie',' +8182-012-7888','casier',TO_DATE('20220112', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(113,1,'Higashihara','Kenwa',22,'barbat','+8165-156-5895','casier',TO_DATE('20220125', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(114,1,'Bando','Yun'A',18,'femeie','+8174-422-3781','femeie/om de serviciu',TO_DATE('20220421', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(115,1,'Kanai','Kuni',30,'barbat','+8147-649-1055','contabil',TO_DATE('20220501', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(120,2,'Petre','Sorin',34,'barbat','+40732-671-544','manager',TO_DATE('20211223', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(121,2,'Popescu','Andrei',52,'barbat','+40736-766-644','curier',TO_DATE('20220107', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(122,2,'Hirdea','Mihaela',33,'femeie','+40725-471-334','casier',TO_DATE('20220227', 'yyyymmdd'));
```

```
INSERT INTO ANGAJAT VALUES
(123,2,'Boboc','Mihai',36,'barbat','+40754-970-123','contabil',TO_DATE('20220201', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(124,2,'Duca','Bianca',23,'femeie','+40722-853-451','administrator',TO_DATE('20220513', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(125,2,'Genes','Dana',19,'femeie','+40745-311-666','casier',TO_DATE('20220220', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(126,2,'Dumitrescu','Oana',21,'femeie','+40747-986-909','asistent manager',TO_DATE('20220617', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(127,2,'Opinca','Radu',27,'barbat','+40744-226-746','secretar/a',TO_DATE('20220227', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(130,3,'Cristaldo','Marandola',29,'femeie','+390315 7479930','manager',TO_DATE('20211209', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(131,3,'Sansone','Calise',39,'femeie','+390360 7042929','curier',TO_DATE('20220209', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(132,3,'Salvatore','Pietro',34,'barbat','+39 0334 7615289','casier',TO_DATE('20220118', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(133,3,'De luca','Rebecca',44,'femeie','+390387 6410034','curier',TO_DATE('20220316', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(134,3,'Battaglia ','Giulio',29,'barbat','+390332 4147151','administrator',TO_DATE('20220518', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(135,3,'Gambone','Simona',38,'femeie','+390386 8429767','contabil',TO_DATE('20220418', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(136,3,'Bocci','Lucia ',47,'femeie','+390321 7988774','secretar/a',TO_DATE('20220404', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(140,4,'Wendell','Astor',29,'barbat','+4904746 96 61 81','manager',TO_DATE('20211225', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(141,4,'Vieth','Reiner',38,'barbat','+4902682 33 18 59','curier',TO_DATE('20220208', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(142,4,'Schulze','Manni',26,'femeie','+4906581 62 50 44','contabil',TO_DATE('20220126', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
(143,4,'Brandt','Fabian',28,'barbat','+4906581 62 50 44','curier',TO_DATE('20220208', 'yyyymmdd'));
INSERT INTO ANGAJAT VALUES
```

```
(144,4,'Schröder','Julie',18,'femeie','+4904746 96 61 81','secretar/a',TO_DATE('20220525', 'yyyymmdd'));

INSERT INTO ANGAJAT VALUES

(145,4,'Busch','Valeria ',19,'femeie','+490911 43 31 23','administrator',TO_DATE('20220414', 'yyyymmdd'));

INSERT INTO ANGAJAT VALUES

(150,5,'Rhys','Lewis',39,'barbat','+44077 2082 4735','manager',TO_DATE('20211215', 'yyyymmdd'));

INSERT INTO ANGAJAT VALUES

(151,5,'Eli','Thomas',45,'barbat','+44078 3886 0633','curier',TO_DATE('20220222', 'yyyymmdd'));

INSERT INTO ANGAJAT VALUES

(152,5,'Bentley','George',24,'barbat','+44070 2301 5654','casier',TO_DATE('20220317', 'yyyymmdd'));

INSERT INTO ANGAJAT VALUES

(153,5,'Coleman','Katy',24,'femeie','+44070 2301 5654','casier',TO_DATE('20220225', 'yyyymmdd'));

INSERT INTO ANGAJAT VALUES

(154,5,'Wade','Edith',21,'femeie','+44070 2301 5654','administrator',TO_DATE('20220326', 'yyyymmdd'));

INSERT INTO ANGAJAT VALUES

(155,5,'Sheldon','Joe ',34,'barbat','+44070 2301 5654','curier',TO_DATE('20220425', 'yyyymmdd'));


SELECT * FROM ANGAJAT;
```

	ID_ANGAJAT	ID_MAGAZIN	NUME_ANGAJAT	PRENUME_ANGAJAT	VARSTA_ANGAJAT	SEX_ANGA
1	110	1	Domen	Tsukiko	23	barbat
2	111	1	Yogi	Toshio	42	barbat
3	112	1	Arima	Yasuo	27	femeie
4	113	1	Higashihara	Kenwa	22	barbat
5	114	1	Bando	Yun'A	18	femeie
6	115	1	Kanai	Kuni	30	barbat
7	120	2	Petre	Sorin	34	barbat
8	121	2	Popescu	Andrei	52	barbat

Sisteme de Gestiune a Bazelor de Date
ANUL II, SERIA 24

----------TABEL SALARII_2022----------

```
CREATE TABLE SALARII2022 (id_angajat NUMBER(5) CONSTRAINT id_angajat REFERENCES ANGAJAT(id_angajat),
                    data_incasare_salariu DATE CONSTRAINT data_incasare_salariu NOT NULL,
                    CONSTRAINT PKEY_SALARII2022 PRIMARY KEY(id_angajat, data_incasare_salariu),
                    plata_cu_ora FLOAT CONSTRAINT plata_cu_ora NOT NULL,
                    ore_lucrate INT CONSTRAINT ore_lucrate NOT NULL,
                    bonus_tip VARCHAR(50),
                    bonus_val FLOAT,
                    salariu_total FLOAT,
                    moneda VARCHAR(50)
                    );


INSERT INTO SALARII2022 VALUES
(110,TO_DATE('20220127', 'yyyymmdd'),16.00,300,'bonus manager',134.78,3989.78,'euro');
INSERT INTO SALARII2022 VALUES
(112,TO_DATE('20220127', 'yyyymmdd'),18.00,230,'',0.0,3002.93,'euro');
INSERT INTO SALARII2022 VALUES
(124,TO_DATE('20220127', 'yyyymmdd'),15.00,290,'',0.0,3009.12,'euro');
INSERT INTO SALARII2022 VALUES
(122,TO_DATE('20220127', 'yyyymmdd'),19.00,180,'',0.0,2897.65,'euro');
INSERT INTO SALARII2022 VALUES
(112,TO_DATE('20220127', 'yyyymmdd'),16.00,301,'',0.0,4002.99,'euro');
INSERT INTO SALARII_2022 VALUES
(121,TO_DATE('20220227', 'yyyymmdd'),18.00,381,'bonus curier',523.32,4552.93,'euro');
INSERT INTO SALARII2022 VALUES
(115,TO_DATE('20220227', 'yyyymmdd'),20.00,271,'',0.0,3452.09,'euro');
INSERT INTO SALARII2022 VALUES
(133,TO_DATE('20220227', 'yyyymmdd'),18.00,345,'bonus curier',457.87,4762.99,'euro');
INSERT INTO SALARII2022 VALUES
(126,TO_DATE('20220327', 'yyyymmdd'),19.00,341,'',0.0,4002.99,'euro');
INSERT INTO SALARII2022 VALUES
```

(123,TO_DATE('20220327', 'yyyymmdd'),20.00,299,'',0.0,4902.29,'euro');

INSERT INTO SALARII2022 VALUES

(132,TO_DATE('20220427', 'yyyymmdd'),17.00,334,'',0.0,3062.99,'euro');

INSERT INTO SALARII2022 VALUES

(127,TO_DATE('20220427', 'yyyymmdd'),17.00,321,'',0.0,3023.49,'euro');

INSERT INTO SALARII2022 VALUES

(136,TO_DATE('20220427', 'yyyymmdd'),17.00,311,'',0.0,4012.11,'euro');

INSERT INTO SALARII2022 VALUES

(120,TO_DATE('20220427', 'yyyymmdd'),16.00,331,'bonus manager',523.99,4402.99,'euro');

INSERT INTO SALARII2022 VALUES

(154,TO_DATE('20220527', 'yyyymmdd'),17.00,309,'',0.0,3212.49,'euro');

INSERT INTO SALARII2022 VALUES

(153,TO_DATE('20220527', 'yyyymmdd'),17.00,315,'',0.0,4032.34,'euro');

SELECT * FROM SALARII2022;

ID_ANGAJAT	DATA_INCASARE_SALARIU	PLATA_CU_ORA	ORE_LUCRATE	BONUS_TIP
110	27-JAN-22	16	300	bonus manag
112	27-JAN-22	18	230	(null)
124	27-JAN-22	15	290	(null)
122	27-JAN-22	19	180	(null)
115	27-FEB-22	20	271	(null)
133	27-FEB-22	18	345	bonus curie
126	27-MAR-22	19	341	(null)
123	27-MAR-22	20	299	(null)
132	27-MAR-22	17	334	(null)

----------TABEL MASINA----------

CREATE TABLE MASINA (id_masina NUMBER(5) CONSTRAINT PKEY_MASINA PRIMARY KEY,

id_curier NUMBER(5),

CONSTRAINT fk_masina FOREIGN KEY (id_curier) REFERENCES ANGAJAT(id_angajat),

nr_matricol VARCHAR(50) CONSTRAINT nr_matricol UNIQUE NOT NULL,

marca_masina VARCHAR(50) CONSTRAINT marca_masina NOT NULL,

tip_masina VARCHAR(50) CONSTRAINT tip_masina NOT NULL);

Sisteme de Gestiune a Bazelor de Date
ANUL II, SERIA 24

INSERT INTO MASINA VALUES

(1,111,'5CE9345','Toyota','mica');

INSERT INTO MASINA VALUES

(2,121,'P67HNU','Nissan','mare');

INSERT INTO MASINA VALUES

(3,131,'8ALB027','Nissan','mare');

INSERT INTO MASINA VALUES

(4,133,'CCDF24','Ford','mica');

INSERT INTO MASINA VALUES

(5,141,'6HQB483','Mitsubishi','mica');

INSERT INTO MASINA VALUES

(6,143,'SDDF483','Ford','mare');

INSERT INTO MASINA VALUES

(7,151,'2SS2D33','Toyota','mare');

INSERT INTO MASINA VALUES

(8,155,'SDSQB23DS','Nissan','mica');

ID_MASINA	ID_CURIER	NR_MATRICOL	MARCA_MASINA	TIP_MASINA
1	111	5CE9345	Toyota	mica
2	121	P67HNU	Nissan	mare
3	131	8ALB027	Nissan	mare
4	133	CCDF24	Ford	mica
5	141	6HQB483	Mitsubishi	mica
6	143	SDDF483	Ford	mare
7	151	2SS2D33	Toyota	mare
8	155	SDSQB23DS	Nissan	mica

SELECT * FROM MASINA;

----------TABEL CLIENT----------

CREATE TABLE CLIENT (id_client NUMBER(5) CONSTRAINT PKEY_CLIENT PRIMARY KEY,

nume_client VARCHAR(50) CONSTRAINT nume_client NOT NULL,

prenume_client VARCHAR(50) CONSTRAINT prenume_client NOT NULL,

mail_client VARCHAR(50) CONSTRAINT mail_client UNIQUE NOT NULL,

telefon_client VARCHAR(50) CONSTRAINT telefon_client NOT NULL,

varsta_client INT CONSTRAINT varsta_client NOT NULL,

data_nastere_client DATE CONSTRAINT data_nastere_client NOT NULL,

adresa_client VARCHAR(100)

);

INSERT INTO CLIENT VALUES

(1,'Merealbe','Briana','bri.mere@yahoo.com','0756234111',20,TO_DATE('20020105', 'yyyymmdd'),'');

```
INSERT INTO CLIENT VALUES
(2,'Buturuga','Patricia','patri_b@gmail.com','0733908002',21,TO_DATE('20010909', 'yyyymmdd'),'');
INSERT INTO CLIENT VALUES
(3,'Mihaila','Robert','mRobertt@yahoo,com','0754333123',15,TO_DATE('20071219', 'yyyymmdd'),'');
INSERT INTO CLIENT VALUES
(4,'Dobrescu','Diana','diaDobre@yahoo.com','0732654986',20,TO_DATE('20020424', 'yyyymmdd'),'');
INSERT INTO CLIENT VALUES
(5,'Furdui','Andrei','furduiandrei@gmail.com','0756123222',19,TO_DATE('20030404', 'yyyymmdd'),'');

SELECT * FROM CLIENT;
```

ID_CLIENT	NUME_CLIENT	PRENUME_CLIENT	MAIL_CLIENT	TELEFON
1	Merealbe	Briana	bri.mere@yahoo.com	07562341
2	Buturuga	Patricia	patri_b@gmail.com	07339080
3	Mihaila	Robert	mRobertt@yahoo,com	07543331
4	Dobrescu	Diana	diaDobre@yahoo.com	07326549
5	Furdui	Andrei	furduiandrei@gmail.com	07561232

----------TABEL PRODUS---------

```
CREATE TABLE PRODUS (id_produs NUMBER(5) CONSTRAINT PKEY_PRODUS PRIMARY KEY,
                     tip_produs VARCHAR(50),
                     genre_produs VARCHAR(50),
                     nume_produs VARCHAR(50) CONSTRAINT nume_produs NOT NULL,
                     pret_unit FLOAT CONSTRAINT pret_unit NOT NULL,
                     moneda_produs VARCHAR(50) CONSTRAINT moneda_produs NOT NULL,
                     descriere_produs VARCHAR(300)
                     );

INSERT INTO PRODUS VALUES
(1111,'manga','action','MY HERO ACADEMIA VOL.28',23.00,'euro','');
INSERT INTO PRODUS VALUES
(1112,'manga','action','TOKYO GHOUL VOL.12',20.99,'euro','');
INSERT INTO PRODUS VALUES
(1113,'manga','action','ATTACK ON TITAN VOL.4',25.00,'euro','');
```

```
INSERT INTO PRODUS VALUES
(1114,'manga','action','ATTACK ON TITAN VOL.32',20.5,'euro','');
INSERT INTO PRODUS VALUES
(1115,'manga','action','DEMON SLAYER VOL.18',19.99,'euro','');
INSERT INTO PRODUS VALUES
(1116,'manga','action','JUJUTSU KAISEN VOL.23',21.00,'euro','');
INSERT INTO PRODUS VALUES
(1117,'manga','slice-of-life','BLUE PERIOD VOL.15',22.8,'euro','');
INSERT INTO PRODUS VALUES
(1118,'manga','sport','HAIKYUU VOL.45',20.5,'euro','');

SELECT * FROM PRODUS;
```

ID_PRODUS	TIP_PRODUS	GENRE_PRODUS	NUME_PRODUS	PRET_UNIT	
1111	manga	action	MY HERO ACADEMIA VOL.28	23	eu
1112	manga	action	TOKYO GHOUL VOL.12	20.99	eu
1113	manga	action	ATTACK ON TITAN VOL.4	25	eu
1114	manga	action	ATTACK ON TITAN VOL.32	20.5	eu
1115	manga	action	DEMON SLAYER VOL.18	19.99	eu
1116	manga	action	JUJUTSU KAISEN VOL.23	21	eu
1117	manga	slice-of-life	BLUE PERIOD VOL.15	22.8	eu
1118	manga	sport	HAIKYUU VOL.45	20.5	eu

----------TABEL VANZARE----------

```
CREATE TABLE VANZARE (id_magazin NUMBER(5),
        CONSTRAINT fk_vanzare FOREIGN KEY (id_magazin) REFERENCES MAGAZIN(id_magazin),
        data_incasare DATE CONSTRAINT data_incasare NOT NULL,
        CONSTRAINT PKEY_VANZARE PRIMARY KEY(id_magazin, data_incasare),
        suma_incasata FLOAT CONSTRAINT suma_incasata NOT NULL
        );

INSERT INTO VANZARE VALUES
(1,TO_DATE('20220103', 'yyyymmdd'),523.03);
INSERT INTO VANZARE VALUES
```

(4,TO_DATE('20220412', 'yyyymmdd'),249.77);

INSERT INTO VANZARE VALUES

(4,TO_DATE('20220531', 'yyyymmdd'),400.78);

INSERT INTO VANZARE VALUES

(3,TO_DATE('20220228', 'yyyymmdd'),333.23);

INSERT INTO VANZARE VALUES

(2,TO_DATE('20220223', 'yyyymmdd'),232.45);

INSERT INTO VANZARE VALUES

(5,TO_DATE('20220312', 'yyyymmdd'),509.01);

INSERT INTO VANZARE VALUES

(1,TO_DATE('20220413', 'yyyymmdd'),322.03);

INSERT INTO VANZARE VALUES

(3,TO_DATE('20220321', 'yyyymmdd'),432.7);

ID_MAGAZIN	DATA_INCASARE	SUMA_INCASATA
1	03-JAN-22	523.03
4	12-APR-22	249.77
4	31-MAY-22	400.78
3	28-FEB-22	333.23
2	23-FEB-22	232.45
5	12-MAR-22	509.01
1	13-APR-22	322.03
3	21-MAR-22	432.7

SELECT * FROM VANZARE;

----------TABEL REDUCERE----------

```
CREATE TABLE REDUCERE (cod_reducere NUMBER(5) CONSTRAINT PKEY_REDUCERE PRIMARY KEY,
                reducere_tip VARCHAR(50) CONSTRAINT reducere_tip NOT NULL,
                reducere_val INT CONSTRAINT reducere_val NOT NULL,
                detalii_reducere VARCHAR(100)
                );
```

INSERT INTO REDUCERE VALUES

(12345,'reducere la orice horror manga',15,'');

INSERT INTO REDUCERE VALUES

(23456,'reducere de vara',30,'');

INSERT INTO REDUCERE VALUES

(34567,'reducere de Craciun',50,'');

INSERT INTO REDUCERE VALUES

(45678,'reducere de Black Friday',75,'');

INSERT INTO REDUCERE VALUES

(56789,'reducere de angajat',20,'');

INSERT INTO REDUCERE VALUES

(67890,'reducere la orice action manga',15,'');

SELECT * FROM REDUCERE;

COD_REDUCERE	REDUCERE_TIP	REDUCERE_VAL	DETALII_REDUC
12345	reducere la orice horror manga	15	(null)
23456	reducere de vara	30	(null)
34567	reducere de Craciun	50	(null)
45678	reducere de Black Friday	75	(null)
56789	reducere de angajat	20	(null)
67890	reducere la orice action manga	15	(null)

----------TABEL SALE----------

CREATE TABLE SALE (id_sale NUMBER(5) CONSTRAINT PKEY_SALE PRIMARY KEY,

id_reducere NUMBER(5),

CONSTRAINT fk_reducere_sale FOREIGN KEY (id_reducere) REFERENCES REDUCERE(cod_reducere),

id_magazin NUMBER(5),

CONSTRAINT fk_magazin_sale FOREIGN KEY (id_magazin) REFERENCES MAGAZIN(id_magazin),

data_inceput DATE CONSTRAINT data_inceput NOT NULL,

data_sfarsit DATE CONSTRAINT data_sfarsit NOT NULL

);

INSERT INTO SALE VALUES

(10000,12345,1,TO_DATE('20220303', 'yyyymmdd'),TO_DATE('20220312', 'yyyymmdd'));

INSERT INTO SALE VALUES

(10123,12345,2,TO_DATE('20220109', 'yyyymmdd'),TO_DATE('20220117', 'yyyymmdd'));

INSERT INTO SALE VALUES

(10369,23456,2,TO_DATE('20220412', 'yyyymmdd'),TO_DATE('20220423', 'yyyymmdd'));

INSERT INTO SALE VALUES

(10615,67890,3,TO_DATE('20220511', 'yyyymmdd'),TO_DATE('20220530', 'yyyymmdd'));

INSERT INTO SALE VALUES

(10123,45678,5,TO_DATE('20220226', 'yyyymmdd'),TO_DATE('20220227', 'yyyymmdd'));

SELECT * FROM SALE;

ID_SALE	ID_REDUCERE	ID_MAGAZIN	DATA_INCEPUT	DATA_SFARSIT
10000	12345	1	03-MAR-22	12-MAR-22
10123	12345	2	09-JAN-22	17-JAN-22
10369	23456	2	12-APR-22	23-APR-22
10615	67890	3	11-MAY-22	30-MAY-22

----------TABEL STOC----------

```
CREATE TABLE STOC (id_stoc NUMBER(5) CONSTRAINT PKEY_STOC PRIMARY KEY,
        id_magazin NUMBER(5),
        CONSTRAINT fk_stoc FOREIGN KEY (id_magazin) REFERENCES MAGAZIN(id_magazin),
        data_livrare DATE CONSTRAINT data_livrare NOT NULL
        );

INSERT INTO STOC VALUES
(0001,1,TO_DATE('20220101', 'yyyymmdd'));
INSERT INTO STOC VALUES
(0002,2,TO_DATE('20220101', 'yyyymmdd'));
INSERT INTO STOC VALUES
(0003,2,TO_DATE('20220212', 'yyyymmdd'));
INSERT INTO STOC VALUES
(0004,3,TO_DATE('20220101', 'yyyymmdd'));
INSERT INTO STOC VALUES
(0005,5,TO_DATE('20220531', 'yyyymmdd'));
INSERT INTO STOC VALUES
(0006,4,TO_DATE('20220516', 'yyyymmdd'));

SELECT * FROM STOC;
```

ID_STOC	ID_MAGAZIN	DATA_LIVRARE
1	1	01-JAN-22
2	2	01-JAN-22
3	2	12-FEB-22
4	3	01-JAN-22
5	5	31-MAY-22
6	4	16-MAY-22

Sisteme de Gestiune a Bazelor de Date
ANUL II, SERIA 24

----------TABEL DETALII STOC----------

```
CREATE TABLE DETALII_STOC (id_stocItem NUMBER(5) CONSTRAINT PKEY_stocITEM PRIMARY KEY,
                    id_stoc NUMBER(5),
                    CONSTRAINT fk_stoc_stocITEM FOREIGN KEY (id_stoc) REFERENCES STOC(id_stoc),
                    id_produs NUMBER(5),
                    CONSTRAINT fk_produs_stocITEM FOREIGN KEY (id_produs) REFERENCES PRODUS(id_produs),
                    cantitate_produs_stoc NUMBER(5) CONSTRAINT cantitate_produs_stoc NOT NULL,
                    nume_produs_stoc VARCHAR(50) CONSTRAINT nume_produs_stoc NOT NULL
                    );


INSERT INTO DETALII_STOC VALUES
(88888,1,1118,23,'HAIKYUU VOL.45');
INSERT INTO DETALII_STOC VALUES
(99999,1,1112,31,'TOKYO GHOUL VOL.12');
INSERT INTO DETALII_STOC VALUES
(44444,2,1118,34,'HAIKYUU VOL.45');
INSERT INTO DETALII_STOC VALUES
(11111,2,1117,22,'BLUE PERIOD VOL.15');
INSERT INTO DETALII_STOC VALUES
(55555,3,1112,43,'TOKYO GHOUL VOL.12');
INSERT INTO DETALII_STOC VALUES
(77777,4,1115,12,'DEMON SLAYER VOL.18');


SELECT * FROM DETALII_STOC;
```

ID_STOCITEM	ID_STOC	ID_PRODUS	CANTITATE_PRODUS_STOC	NUME_PRODUS_STOC
88888	1	1118	23	HAIKYUU VOL.45
99999	1	1112	31	TOKYO GHOUL VOL.12
44444	2	1118	34	HAIKYUU VOL.45
11111	2	1117	22	BLUE PERIOD VOL.15
55555	3	1112	43	TOKYO GHOUL VOL.12
77777	4	1115	12	DEMON SLAYER VOL.18

Sisteme de Gestiune a Bazelor de Date
ANUL II, SERIA 24

----------TABEL COMANDA----------

```
CREATE TABLE COMANDA (id_comanda NUMBER(5) CONSTRAINT PKEY_COMANDA PRIMARY KEY,

        id_client NUMBER(5), CONSTRAINT fk_client_comanda FOREIGN KEY (id_client) REFERENCES CLIENT(id_client),

        id_magazin NUMBER(5), CONSTRAINT fk_magazin_comanda FOREIGN KEY (id_magazin) REFERENCES
MAGAZIN(id_magazin),

        id_curier NUMBER(5), CONSTRAINT fk_curier_comanda FOREIGN KEY (id_curier) REFERENCES ANGAJAT(id_angajat),

        id_masina NUMBER(5), CONSTRAINT fk_masina_comanda FOREIGN KEY (id_masina) REFERENCES
MASINA(id_masina),

        data_plasare DATE CONSTRAINT data_plasare NOT NULL,

        data_sosire DATE,

        nr_produse INT CONSTRAINT nr_produse NOT NULL,

        suma_initiala FLOAT CONSTRAINT suma_initiala NOT NULL,

        cod_reducere NUMBER(5), CONSTRAINT fk_reducere_comanda FOREIGN KEY (cod_reducere) REFERENCES
REDUCERE(cod_reducere),

        suma_finala FLOAT CONSTRAINT suma_finala NOT NULL,

        moneda_comanda VARCHAR(50) CONSTRAINT moneda_comanda NOT NULL,

        status_comanda VARCHAR(50) CONSTRAINT status_comanda NOT NULL,

        adresa VARCHAR(100) CONSTRAINT adresa NOT NULL

        );


INSERT INTO COMANDA VALUES

(1122,1,2,151,7,TO_DATE('20220109', 'yyyymmdd'),TO_DATE('20220115', 'yyyymmdd'),2,45.55,12345,32.34,'euro','livrata','Std
Stefan cel Mare nr 45, Bacau');

INSERT INTO COMANDA VALUES

(1133,3,2,111,1,TO_DATE('20220302', 'yyyymmdd'),TO_DATE('20220314', 'yyyymmdd'),3,76.09,56789,54,'euro','livrata','Std
Stefan cel Mare nr 45, Bacau');

INSERT INTO COMANDA VALUES

(1144,3,3,131,3,TO_DATE('20220423', 'yyyymmdd'),TO_DATE('20220514', 'yyyymmdd'),5,130.98,23456,121.43,'euro','livrata','Std
Danescu Negru nr 2, Buzau');

INSERT INTO COMANDA VALUES

(1155,4,4,141,5,TO_DATE('20220619', 'yyyymmdd'),TO_DATE('20220620', 'yyyymmdd'),3,65.3,34567,56.99,'euro','livrata','Std
Floriilor nr 5, Iasi');

INSERT INTO COMANDA VALUES

(1166,2,1,151,7,TO_DATE('20220530', 'yyyymmdd'),TO_DATE('', 'yyyymmdd'),2,43.99,12345,38.17,'euro','in curs de livrare','Std
Stefan cel Mare nr 23, Bacau');
```

Sisteme de Gestiune a Bazelor de Date
ANUL II, SERIA 24

SELECT * FROM COMANDA;

ID_COMANDA	ID_CLIENT	ID_MAGAZIN	ID_CURIER	ID_MASINA	DATA_PLASARE	DATA_SOSIRE	NR_PR(
1166	2	1	151	7	30-MAY-22	(null)	
1122	1	2	151	7	09-JAN-22	15-JAN-22	
1133	3	2	111	1	02-MAR-22	14-MAR-22	
1144	3	3	131	3	23-APR-22	14-MAY-22	
1155	4	4	141	5	19-JUN-22	20-JUN-22	
1177	3	2	111	1	02-MAR-22	14-MAR-22	

----------TABEL DETALII COMANDA--------

CREATE TABLE DETALII_COMANDA (id_comandaItem NUMBER(5) CONSTRAINT PKEY_comandaITEM PRIMARY KEY,

id_comanda NUMBER(5), CONSTRAINT fk_comanda_comandaITEM FOREIGN KEY (id_comanda) REFERENCES COMANDA(id_comanda),

id_produs NUMBER(5), CONSTRAINT fk_produs_comandaITEM FOREIGN KEY (id_produs) REFERENCES PRODUS(id_produs),

cantitate INT CONSTRAINT cantitate NOT NULL

);

INSERT INTO DETALII_COMANDA VALUES

(987,1166,1111,1);

INSERT INTO DETALII_COMANDA VALUES

(876,1155,1113,2);

INSERT INTO DETALII_COMANDA VALUES

(765,1122,1116,1);

INSERT INTO DETALII_COMANDA VALUES

(654,1122,11104,1);

INSERT INTO DETALII_COMANDA VALUES

(543,1144,1118,1);

ID_COMANDAITEM	ID_COMANDA	ID_PRODUS	CANTITATE
987	1166	1111	1
876	1155	1113	2
765	1122	1116	1
543	1144	1118	1

SELECT * FROM DETALII_COMANDA;

6. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent care să utilizeze două tipuri diferite de colecții studiate. Apelați subprogramul.

----------EX6--------

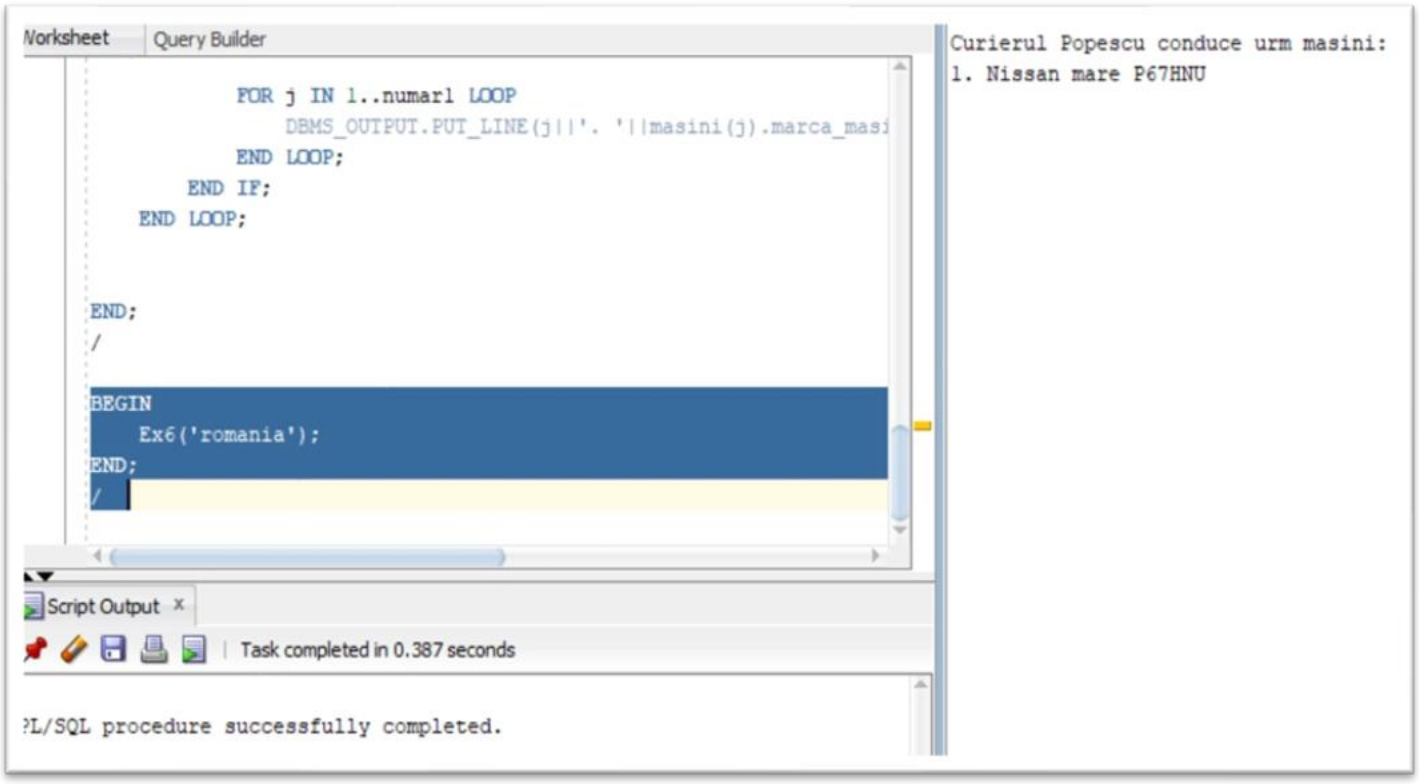
----pentru un magazin al carei tara este data vrem sa afisam pentru fiecare curier

----ce masini conduc:

```
CREATE OR REPLACE PROCEDURE Ex6 (tara MAGAZIN.tara_magazin%TYPE)
AS
   TYPE tabel_idx IS TABLE OF masina%rowtype INDEX BY PLS_INTEGER;
   masini tabel_idx;

   TYPE angajat_rec is RECORD(
   id_angajat angajat.id_angajat%TYPE,
   id_magazin angajat.id_magazin%TYPE,
   nume angajat.nume_angajat%TYPE,
   prenume angajat.prenume_angajat%TYPE,
   functie angajat.functie%TYPE);

   TYPE my_nested_table IS TABLE OF angajat_rec;
   angajati my_nested_table := my_nested_table();

   numar NUMBER(6);
   numar1 NUMBER(6);

BEGIN

   SELECT COUNT(*)
   INTO numar
```

```
  FROM angajat
  WHERE functie = 'curier';


  angajati.extend(numar + 1);


  SELECT a.id_angajat,a.id_magazin, a.nume_angajat, a.prenume_angajat, a.functie
  BULK COLLECT INTO angajati
  FROM angajat a, magazin s
  WHERE a.functie = 'curier' and a.id_magazin=s.id_magazin AND UPPER(s.tara_magazin) LIKE  UPPER(tara);


  FOR i IN angajati.first..angajati.last LOOP
    DBMS_OUTPUT.PUT_LINE('Curierul '||angajati(i).nume||' conduce urm masini:');


    SELECT COUNT(*)
    INTO numar1
    FROM masina m
    WHERE m.id_curier = angajati(i).id_angajat;


    IF numar1 = 0 THEN
      DBMS_OUTPUT.PUT_LINE('nu are inca masina');
    END IF;


    IF numar1 > 0 THEN
      SELECT *
      BULK COLLECT INTO masini
      FROM masina m
      WHERE m.id_curier = angajati(i).id_angajat;


      FOR j IN 1..numar1 LOOP
        DBMS_OUTPUT.PUT_LINE(j||'. '||masini(j).marca_masina||' '||masini(j).tip_masina||' '||masini(j).nr_matricol);
      END LOOP;
    END IF;
  END LOOP;
```

```
END;

/

BEGIN

   Ex6('romania');

END;

/
```



7. 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. Apelați subprogramul.

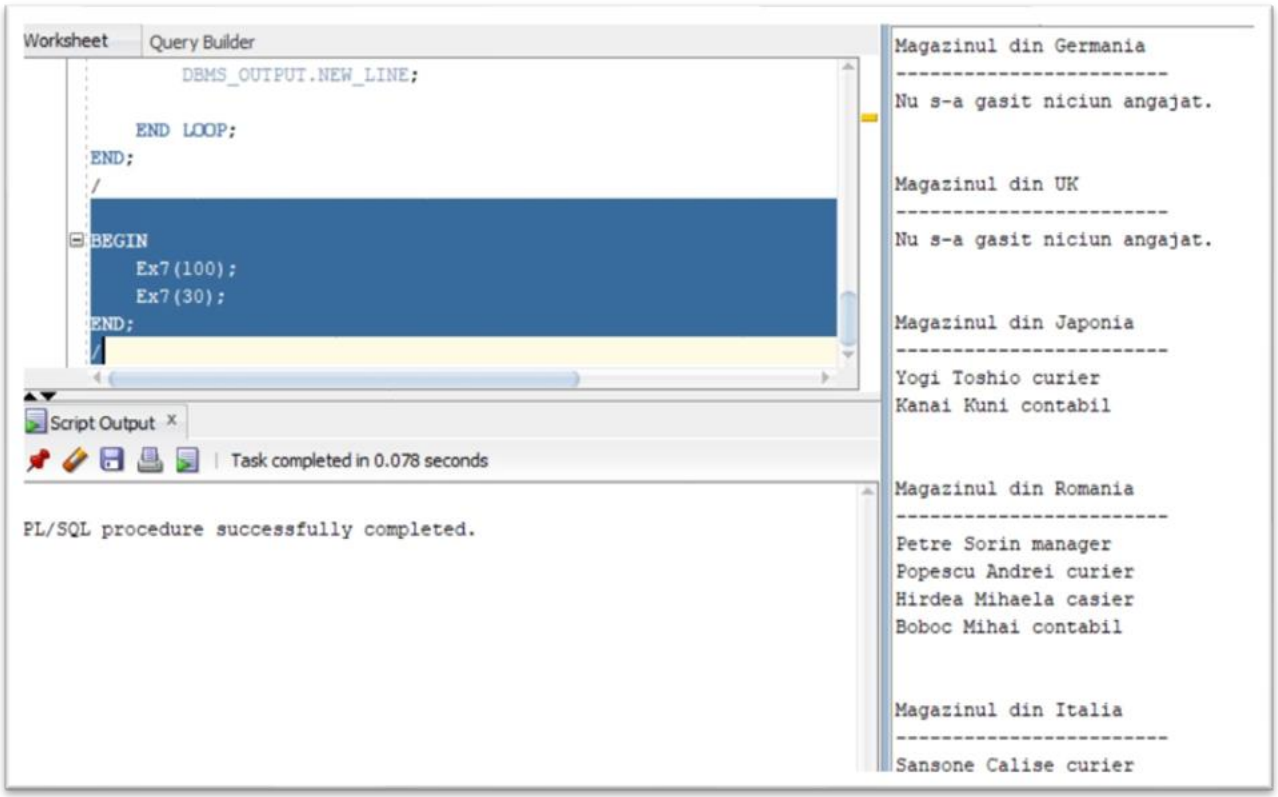
----------EX7--------

----sa se afiseze pentru fiecare magazin angajatii care au peste varsta data

----daca nu exista angajati care sa aibe peste varsta respectiva se afiseaza un mesaj

```
CREATE OR REPLACE PROCEDURE Ex7 (age angajat.varsta_angajat%TYPE)
AS
   CURSOR magazine IS
      SELECT tara_magazin, id_magazin
      FROM MAGAZIN;
   m magazine%ROWTYPE;
   CURSOR angajati(id_m MAGAZIN.ID_MAGAZIN%TYPE) IS
            SELECT nume_angajat, prenume_angajat, functie
            FROM ANGAJAT a
            WHERE a.varsta_angajat >= age AND a.id_magazin = id_m;
   a angajati%ROWTYPE;
   numar NUMBER(6);
BEGIN
   FOR m in magazine LOOP
      DBMS_OUTPUT.PUT_LINE( 'Magazinul din ' || m.tara_magazin);
      DBMS_OUTPUT.PUT_LINE('------------------------');

      UPDATE angajat
      SET varsta_angajat = varsta_angajat + 0
      WHERE angajat.varsta_angajat >= age and angajat.id_magazin = m.id_magazin;
      if SQL%FOUND THEN
      FOR a in angajati(m.id_magazin) LOOP
```

```
            DBMS_OUTPUT.PUT_LINE( a.nume_angajat || ' ' || a.prenume_angajat || ' ' || a.functie);

            END LOOP;

        ELSE DBMS_OUTPUT.PUT_LINE('Nu s-a gasit niciun angajat.');

        END IF;

        DBMS_OUTPUT.NEW_LINE;

        DBMS_OUTPUT.NEW_LINE;

    END LOOP;

END;

/

BEGIN

    Ex7(100);

    Ex7(30);

END;

/
```



8. **Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent de tip functie care să utilizeze într-o singură comandă SQL3 dintre tabelele definite. Definiți minim 2 excepții. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate.**

----------EX8----------

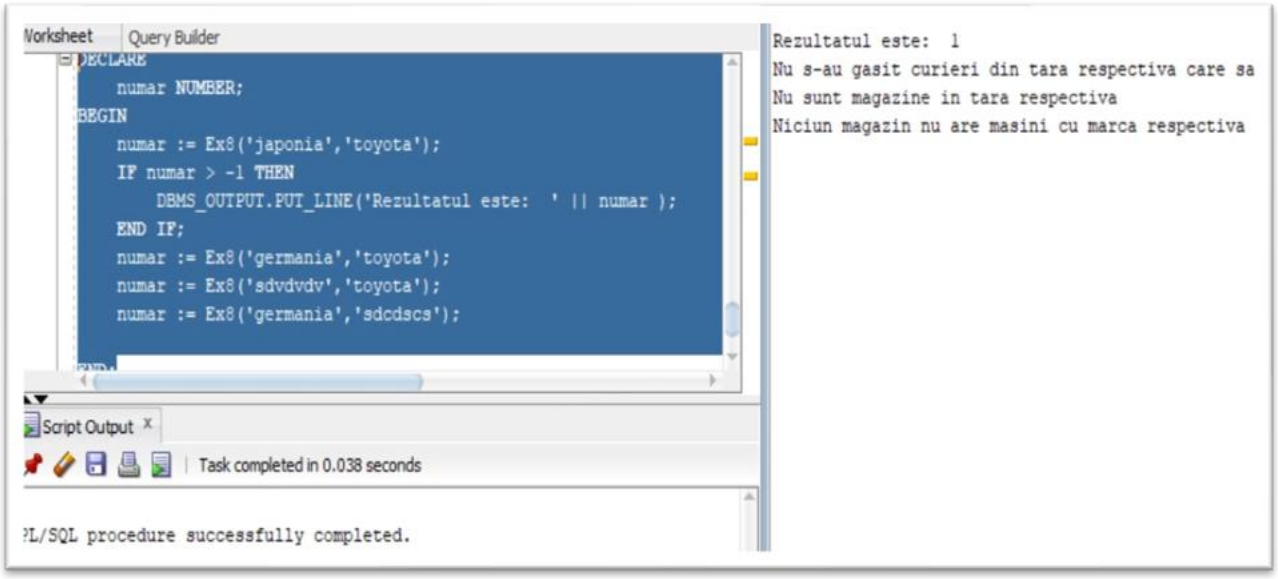
-------pentru o tara si o marca data sa se afiseze cati curieri care lucreaza

-------pt magazinul din tara data conduc o masina de marca respectiva

```
CREATE OR REPLACE FUNCTION Ex8(tara magazin.tara_magazin%TYPE, marca masina.marca_masina%TYPE) RETURN NUMBER
IS
   nr_curieri NUMBER;
   TYPE tab1 is table of magazin%ROWTYPE INDEX BY PLS_INTEGER;
   t tab1;
   TYPE tab2 is table of masina%ROWTYPE INDEX BY PLS_INTEGER;
   m tab2;
   --exceptii
   TARA_DOESNT_EXIST EXCEPTION;
   MARCA_DOESNT_EXIST EXCEPTION;
   NO_DATA_FOUND EXCEPTION;

BEGIN
   SELECT *
   BULK COLLECT INTO t
   FROM magazin m
   WHERE upper(m.tara_magazin) like upper(tara);

   IF SQL%NOTFOUND THEN
      RAISE TARA_DOESNT_EXIST;
   END IF;
```

```
SELECT *
BULK COLLECT INTO m
FROM masina m
WHERE upper(m.marca_masina) like upper(marca);

IF SQL%NOTFOUND THEN
   RAISE MARCA_DOESNT_EXIST;
END IF;

select count(nume_angajat)
into nr_curieri
from angajat a
join magazin m on a.id_magazin = m.id_magazin
join masina n on n.id_curier = a.id_angajat
where upper(m.tara_magazin) like upper(tara) and upper(n.marca_masina) like upper(marca);

IF nr_curieri = 0  THEN
   RAISE NO_DATA_FOUND;
ELSE RETURN nr_curieri;
END IF;

EXCEPTION
   WHEN TARA_DOESNT_EXIST THEN
      DBMS_OUTPUT.PUT_LINE('Nu sunt magazine in tara respectiva');
      RETURN -1;
   WHEN MARCA_DOESNT_EXIST THEN
      DBMS_OUTPUT.PUT_LINE('Niciun magazin nu are masini cu marca respectiva');
      RETURN -1;
   WHEN NO_DATA_FOUND THEN
      DBMS_OUTPUT.PUT_LINE('Nu s-au gasit curieri din tara respectiva care sa conduca masini de marca respectiva');
      RETURN -1;
   WHEN OTHERS THEN
      DBMS_OUTPUT.PUT_LINE('Codul de eroare: ' || SQLCODE);
```

```
        DBMS_OUTPUT.PUT_LINE('Mesajul erorii: ' || SQLERRM);

        RETURN -1;

END;

/

DECLARE

   numar NUMBER;

BEGIN

   numar := Ex8('japonia','toyota');

   IF numar > -1 THEN

      DBMS_OUTPUT.PUT_LINE('Rezultatul este:  ' || numar );

   END IF;

   numar := Ex8('germania','toyota');

   numar := Ex8('sdvdvdv','toyota');

   numar := Ex8('germania','sdcdscs');

END;

/
```



9. **Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent de tip procedură care să utilizeze într-o singură comandă SQL 5 dintre tabelele definite. Tratați toate excepțiile care pot apărea, incluzând excepțiile NO_DATA_FOUND și TOO_MANY_ROWS. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate.**

```
INSERT INTO COMANDA VALUES
(1177,3,2,111,1,TO_DATE('20220302', 'yyyymmdd'),TO_DATE('20220314', 'yyyymmdd'),3,76.09,56789,54,'euro','livrata','Std Stefan cel Mare nr 45, Bacau');
select * from client;
select * from comanda;
```

----------EX9----------

---------pentru un nume dat si o data sa se afiseze numele si prenumele clientului,

---------data plasarii, magazinul, curierul si nr matricol al masinii

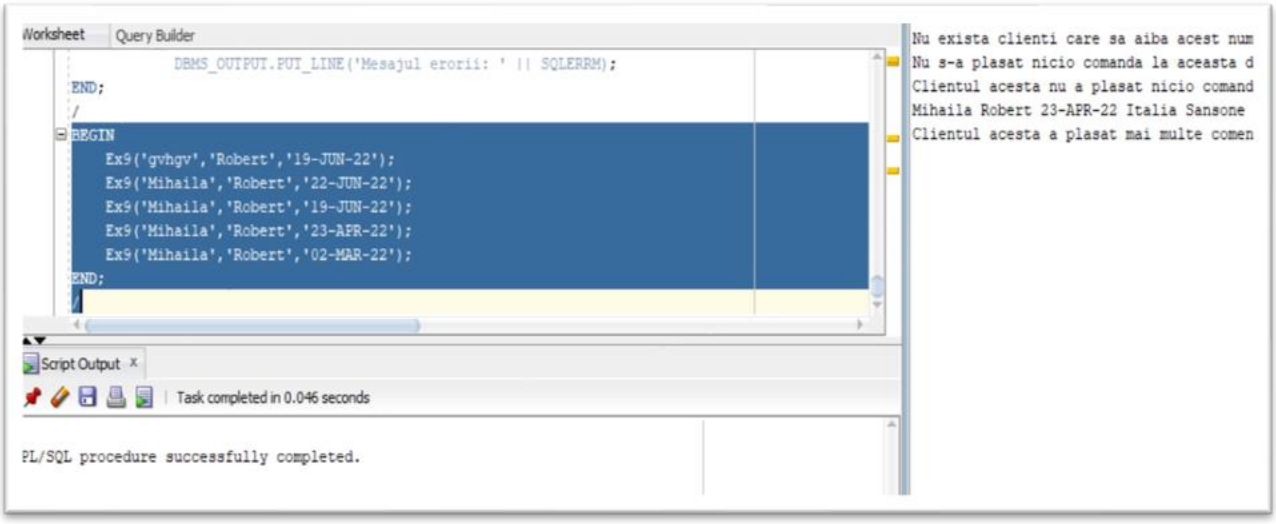
---------de comanda plasata la data respectiva de catre clientul respectiv

```
CREATE OR REPLACE PROCEDURE Ex9(nume client.nume_client%type, prenume client.prenume_client%type, data comanda.data_plasare%type)
AS
   TYPE table1 IS TABLE OF client%rowtype INDEX BY PLS_INTEGER;
   tab1 table1;
   TYPE table2 IS TABLE OF comanda%rowtype INDEX BY PLS_INTEGER;
   tab2 table2;
   TYPE table3 IS TABLE OF comanda.id_comanda%type INDEX BY PLS_INTEGER;
   tab3 table3;

   CURSOR comanda_detalii is
      select nume_client, prenume_client, data_plasare, tara_magazin, nume_angajat, prenume_angajat, nr_matricol
      from comanda co
      join client cl on (co.id_client = cl.id_client)
      join magazin m on (co.id_magazin = m.id_magazin)
```

```
    join angajat a on (co.id_curier = a.id_angajat)
    join masina ms on (co.id_masina = ms.id_masina);

  temp comanda_detalii%rowtype;

  NO_DATA_FOUND1 EXCEPTION;
  NO_DATA_FOUND2 EXCEPTION;
  NO_DATA_FOUND3 EXCEPTION;
  TOO_MANY_ROWS1 EXCEPTION;
  TOO_MANY_ROWS2 EXCEPTION;

BEGIN
  SELECT *
  BULK COLLECT INTO tab1
  FROM client c
  WHERE UPPER(c.nume_client) = UPPER(nume) AND UPPER(c.prenume_client) = UPPER(prenume)
  IF SQL%NOTFOUND THEN
    RAISE NO_DATA_FOUND1;
  END IF;
  IF tab1.count >= 2 THEN
    RAISE TOO_MANY_ROWS1;
  END IF;
  SELECT *
  BULK COLLECT INTO tab2
  FROM comanda c
  WHERE c.data_plasare = data;

 IF SQL%NOTFOUND THEN
    RAISE NO_DATA_FOUND2;
  END IF;
  SELECT id_comanda
  BULK COLLECT INTO tab3
  FROM comanda c
```

```
  JOIN client cl ON (c.id_client = cl.id_client)
  WHERE c.data_plasare = data AND UPPER(cl.nume_client) = UPPER(nume) AND UPPER(cl.prenume_client) = UPPER(prenume);
  IF SQL%NOTFOUND THEN
    RAISE NO_DATA_FOUND3;
  END IF;
  IF tab3.count >= 2 THEN
    RAISE TOO_MANY_ROWS2;
  END IF;
  select nume_client, prenume_client, data_plasare, tara_magazin, nume_angajat, prenume_angajat, nr_matricol
  into temp
  from comanda co
  join client cl on (co.id_client = cl.id_client)
  join magazin m on (co.id_magazin = m.id_magazin)
  join angajat a on (co.id_curier = a.id_angajat)
  join masina ms on (co.id_masina = ms.id_masina)
  where upper(cl.nume_client) like upper(nume) and upper(cl.prenume_client) like upper(prenume) and co.data_plasare = data;
  IF SQL%NOTFOUND THEN
    RAISE NO_DATA_FOUND3;
  END IF;
  DBMS_OUTPUT.PUT_LINE(temp.nume_client|| ' ' || temp.prenume_client || ' ' || temp.data_plasare || ' ' ||
temp.tara_magazin || ' ' || temp.nume_angajat || ' ' || temp.prenume_angajat || ' ' || temp.nr_matricol);
  EXCEPTION
    WHEN NO_DATA_FOUND1 THEN
      DBMS_OUTPUT.PUT_LINE('Nu exista clienti care sa aiba acest nume!');
    WHEN NO_DATA_FOUND2 THEN
      DBMS_OUTPUT.PUT_LINE('Nu s-a plasat nicio comanda la aceasta data!');
    WHEN NO_DATA_FOUND3 THEN
      DBMS_OUTPUT.PUT_LINE('Clientul acesta nu a plasat nicio comanda la data data!');
    WHEN TOO_MANY_ROWS1 THEN
      DBMS_OUTPUT.PUT_LINE('Exista mai multi clienti cu acest nume!');
    WHEN TOO_MANY_ROWS2 THEN
      DBMS_OUTPUT.PUT_LINE('Clientul acesta a plasat mai multe comenzi la data respectiva!');
    WHEN OTHERS THEN
```

```
        DBMS_OUTPUT.PUT_LINE('Codul de eroare: ' || SQLCODE);

        DBMS_OUTPUT.PUT_LINE('Mesajul erorii: ' || SQLERRM);

END;

/

BEGIN

   Ex9('gvhgv','Robert','19-JUN-22');

   Ex9('Mihaila','Robert','22-JUN-22');

   Ex9('Mihaila','Robert','19-JUN-22');

   Ex9('Mihaila','Robert','23-APR-22');

   Ex9('Mihaila','Robert','02-MAR-22');

END;

/
```



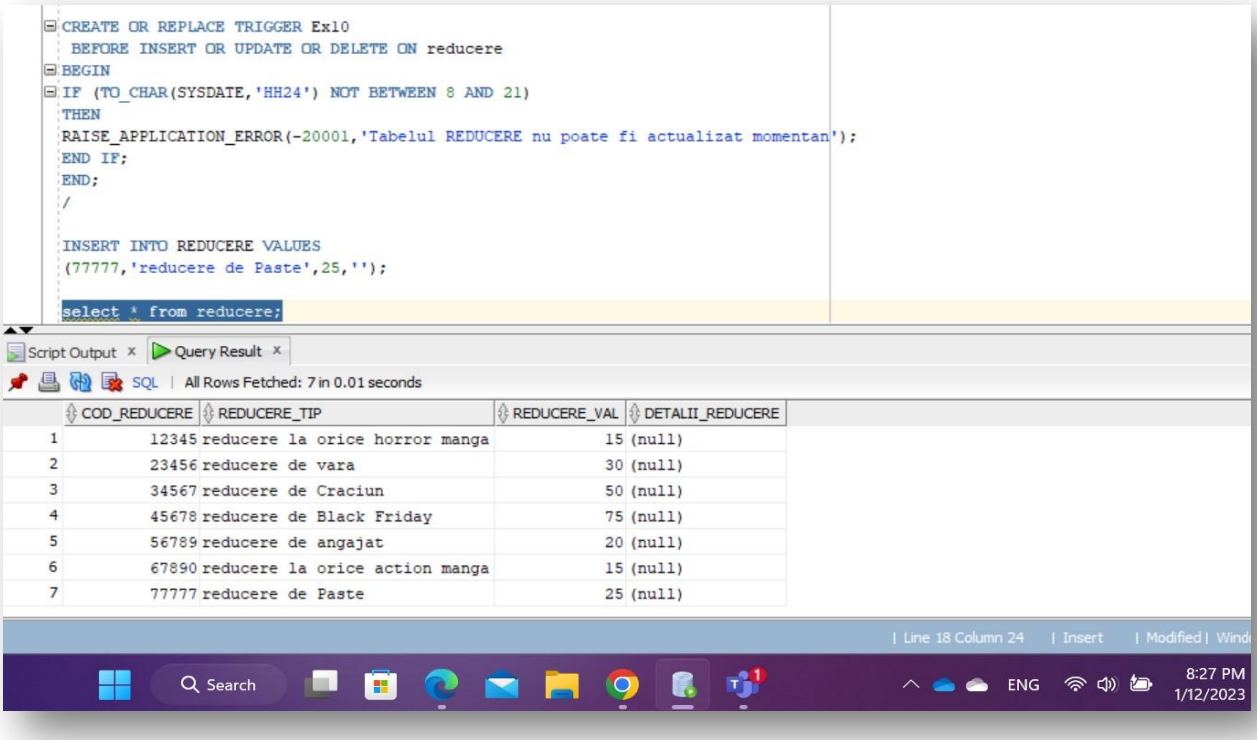
10. Definiți un trigger de tip LMD la nivel de comandă. Declanșați trigger-ul.

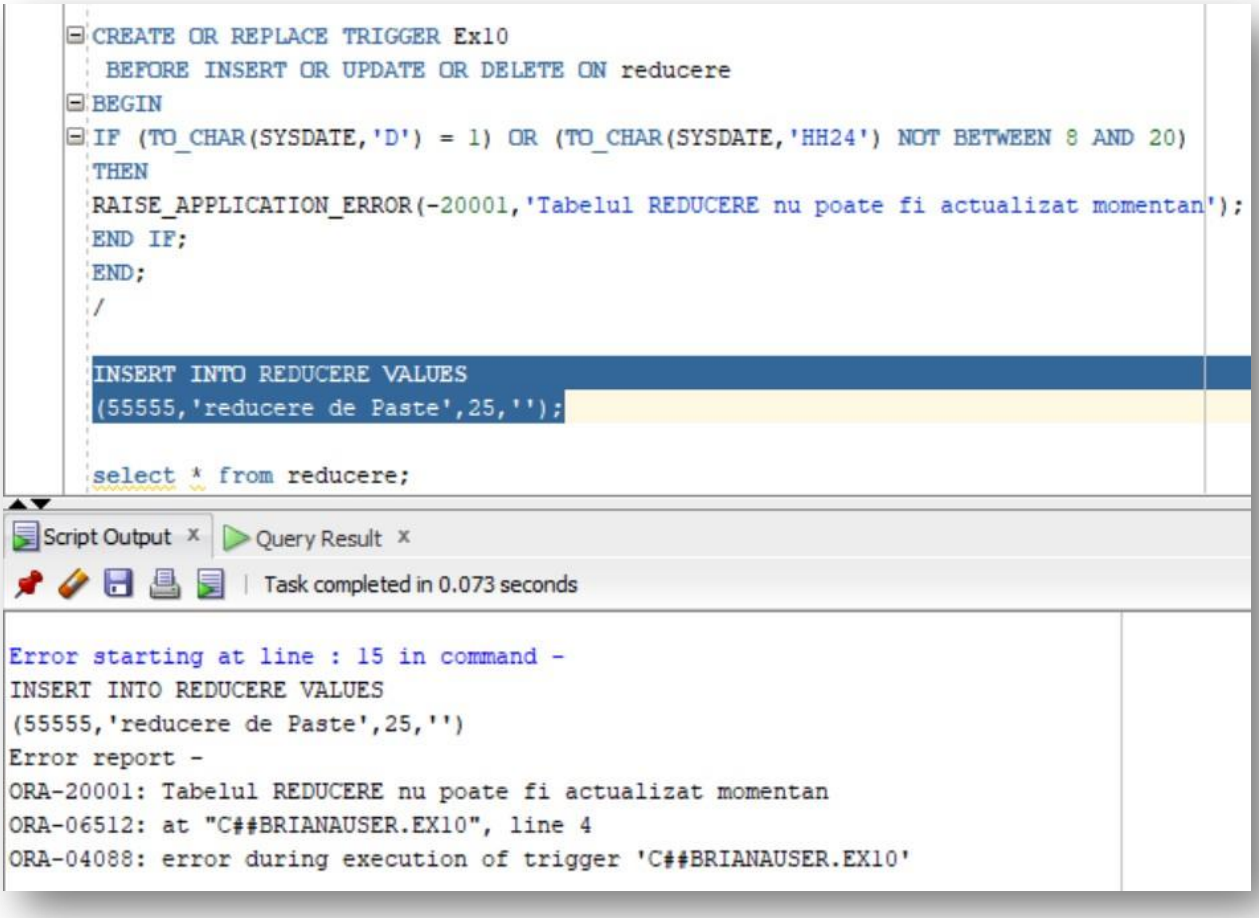
----------EX10------

------Trigger care nu permite actualizarea tabelului REDUCERE

------decat de la 8:00 pana la 21:00, de luni pana sambata

```
CREATE OR REPLACE TRIGGER Ex10

 BEFORE INSERT OR UPDATE OR DELETE ON reducere

BEGIN

IF (TO_CHAR(SYSDATE,'D') = 1) OR (TO_CHAR(SYSDATE,'HH24') NOT BETWEEN 8 AND 20)

THEN

RAISE_APPLICATION_ERROR(-20001,'Tabelul REDUCERE nu poate fi actualizat momentan');

END IF;

END;

/

DROP TRIGGER Ex10;
```





```
CREATE OR REPLACE TRIGGER Ex10
  BEFORE INSERT OR UPDATE OR DELETE ON reducere
BEGIN
IF (TO_CHAR(SYSDATE,'D') = 1) OR (TO_CHAR(SYSDATE,'HH24') NOT BETWEEN 8 AND 20)
THEN
RAISE_APPLICATION_ERROR(-20001,'Tabelul REDUCERE nu poate fi actualizat momentan');
END IF;
END;
/

INSERT INTO REDUCERE VALUES
(55555,'reducere de Paste',25,'');

select * from reducere;
```

Script Output x | Query Result x

Task completed in 0.073 seconds

```
Error starting at line : 15 in command -
INSERT INTO REDUCERE VALUES
(55555,'reducere de Paste',25,'')
Error report -
ORA-20001: Tabelul REDUCERE nu poate fi actualizat momentan
ORA-06512: at "C##BRIANAUSER.EX10", line 4
ORA-04088: error during execution of trigger 'C##BRIANAUSER.EX10'
```

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

INSERT INTO COMANDA VALUES

(9999,3,2,111,1,TO_DATE('20220302', 'yyyymmdd'),TO_DATE('20220314', 'yyyymmdd'),3,76.09,56789,54,'euro','livrata','Std Stefan cel Mare nr 45, Bacau');

select * from comanda;

INSERT INTO DETALII_COMANDA VALUES

(999,9999,1118,1);

select * from detalii_comanda;

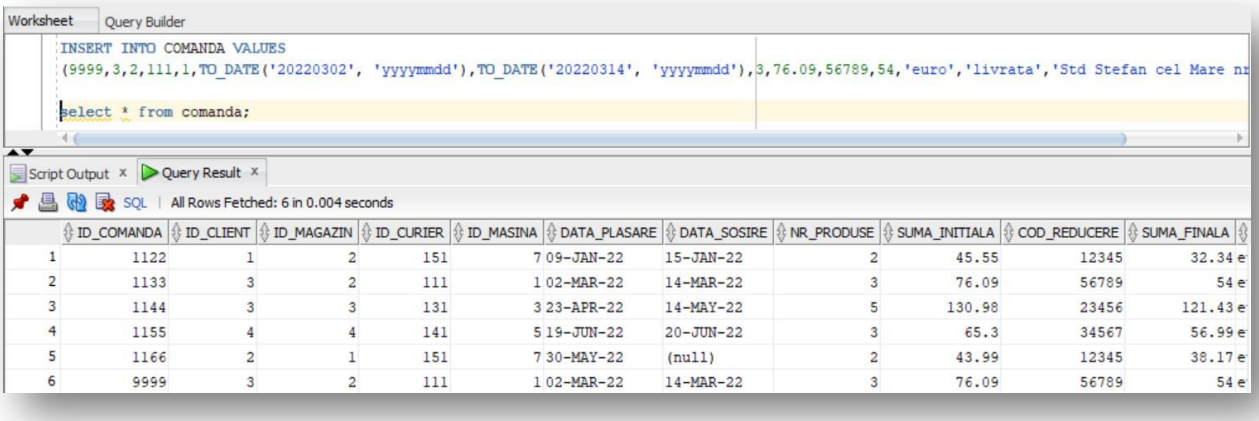
Sisteme de Gestiune a Bazelor de Date
ANUL II, SERIA 24

----------EX11----------

------Trigger care sterge automat toate randurile din tabelul detalii_comanda

------care au legatura cu o comanda stearsa din bd

```
CREATE OR REPLACE TRIGGER Ex11
AFTER DELETE ON comanda FOR EACH ROW
declare
   id_temp number;
begin
   id_temp := :OLD.id_comanda;
   delete from detalii_comanda where id_comanda = id_temp;
END;

DELETE FROM comanda WHERE id_comanda = 9999;
select * from comanda;
select * from detalii_comanda;
```



	ID_COMANDA	ID_CLIENT	ID_MAGAZIN	ID_CURIER	ID_MASINA	DATA_PLASARE	DATA_SOSIRE	NR_PRODUSE	SUMA_INITIALA	COD_REDUCERE	SUMA_FINALA
1	1122	1	2	151	7	09-JAN-22	15-JAN-22	2	45.55	12345	32.34 e
2	1133	3	2	111	1	02-MAR-22	14-MAR-22	3	76.09	56789	54 e
3	1144	3	3	131	3	23-APR-22	14-MAY-22	5	130.98	23456	121.43 e
4	1155	4	4	141	5	19-JUN-22	20-JUN-22	3	65.3	34567	56.99 e
5	1166	2	1	151	7	30-MAY-22	(null)	2	43.99	12345	38.17 e
6	9999	3	2	111	1	02-MAR-22	14-MAR-22	3	76.09	56789	54 e

Sisteme de Gestiune a Bazelor de Date
ANUL II, SERIA 24

```
INSERT INTO DETALII_COMANDA VALUES
(999,9999,1118,1);

select * from detalii_comanda;
```

Script Output | Query Result

SQL | All Rows Fetched: 6 in 0.003 seconds

	ID_COMANDAITEM	ID_COMANDA	ID_PRODUS	CANTITATE
1	987	1166	1111	1
2	876	1155	1113	2
3	765	1122	1116	1
4	543	1144	1118	1
5	654	1122	1114	1
6	999	9999	1118	1

```
CREATE OR REPLACE TRIGGER Ex11
AFTER DELETE ON comanda FOR EACH ROW
declare
    id_temp number;
begin
    id_temp := :OLD.id_comanda;
    delete from detalii_comanda where id_comanda = id_temp;
END;

DELETE FROM comanda WHERE id_comanda = 9999;
select * from comanda;
```

Script Output | Query Result | Query Result 1 | Query Result 2

SQL | All Rows Fetched: 5 in 0.008 seconds

	ID_COMANDA	ID_CLIENT	ID_MAGAZIN	ID_CURIER	ID_MASINA	DATA_PLASARE
1	1122	1	2	151	7	09-JAN-22
2	1133	3	2	111	1	02-MAR-22
3	1144	3	3	131	3	23-APR-22
4	1155	4	4	141	5	19-JUN-22
5	1166	2	1	151	7	30-MAY-22

```
CREATE OR REPLACE TRIGGER Ex11
AFTER DELETE ON comanda FOR EACH ROW
declare
    id_temp number;
begin
    id_temp := :OLD.id_comanda;
    delete from detalii_comanda where id_comanda = id_temp;
END;

DELETE FROM comanda WHERE id_comanda = 9999;
select * from comanda;
select * from detalii_comanda;
```

Script Output | Query Result | Query Result 1 | Query Result 2

SQL | All Rows Fetched: 5 in 0.003 seconds

	ID_COMANDAITEM	ID_COMANDA	ID_PRODUS	CANTITATE
1	987	1166	1111	1
2	876	1155	1113	2
3	765	1122	1116	1
4	543	1144	1118	1
5	654	1122	1114	1

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

----------EX12----------

-------Trigger de tip LDD care se declanseaza de fiecare data cand se executa operatii

-------ALTER, DROP SAU CREATE

-------pentru a testa daca merge vom stoca informatiile intr-un tabel separat

```
CREATE TABLE LDD_info(  utilizator VARCHAR2(50),
                baza_de_date VARCHAR2(50),
                eveniment VARCHAR2(50),
                nume_obj VARCHAR2(50),
                tip_obj VARCHAR2(50),
                data_realizare DATE
              );
SELECT * FROM LDD_info;
```

Sisteme de Gestiune a Bazelor de Date
ANUL II, SERIA 24

```
CREATE OR REPLACE TRIGGER Ex12

   AFTER CREATE OR ALTER OR DROP ON SCHEMA

BEGIN

   INSERT INTO LDD_info

   VALUES(SYS.LOGIN_USER, SYS.DATABASE_NAME, SYS.SYSEVENT, SYS.DICTIONARY_OBJ_NAME, SYS.DICTIONARY_OBJ_TYPE,
SYSDATE);

END;

/

CREATE TABLE LOCUINTE ( id_locuinta NUMBER(6) CONSTRAINT id_locuinta_pk PRIMARY KEY,

                nume_proprietar VARCHAR2(100),

                oras VARCHAR2(100));

select * from LOCUINTE;

ALTER TABLE LOCUINTE

ADD (prenume_proprietar VARCHAR2(100));

ALTER TABLE LOCUINTE

DROP COLUMN prenume_proprietar;

DROP TABLE LOCUINTE;

SELECT * FROM LDD_info;
```