

**Proiect la disciplina**  
**Elemente avansate de baze de date**

**Domeniu: Brutărie**

**Popoviciu Andreea**  
**Informatică Economică**  
**An III, grupa 5**  
**2022-2023**

## Scenariu

Pentru a menține o colaborare eficientă între magazinele proprii și lanțurile de magazine partenere ale unei brutării, dar și pentru a oferi produse mereu proaspete, de înaltă calitate, aceasta are nevoie de o bază de date.

Brutăria este situată în Sighișoara și deține magazine proprii în 3 localități din județul Mureș, Sighișoara, Sovata și Tg. Mureș.

În cadrul brutăriei se realizează produse de panificație, patiserie și specialități care sunt distribuite prin magazinele proprii sau magazinele partenere. Pentru fiecare produs se va specifica codul, denumirea și tipul acestuia.

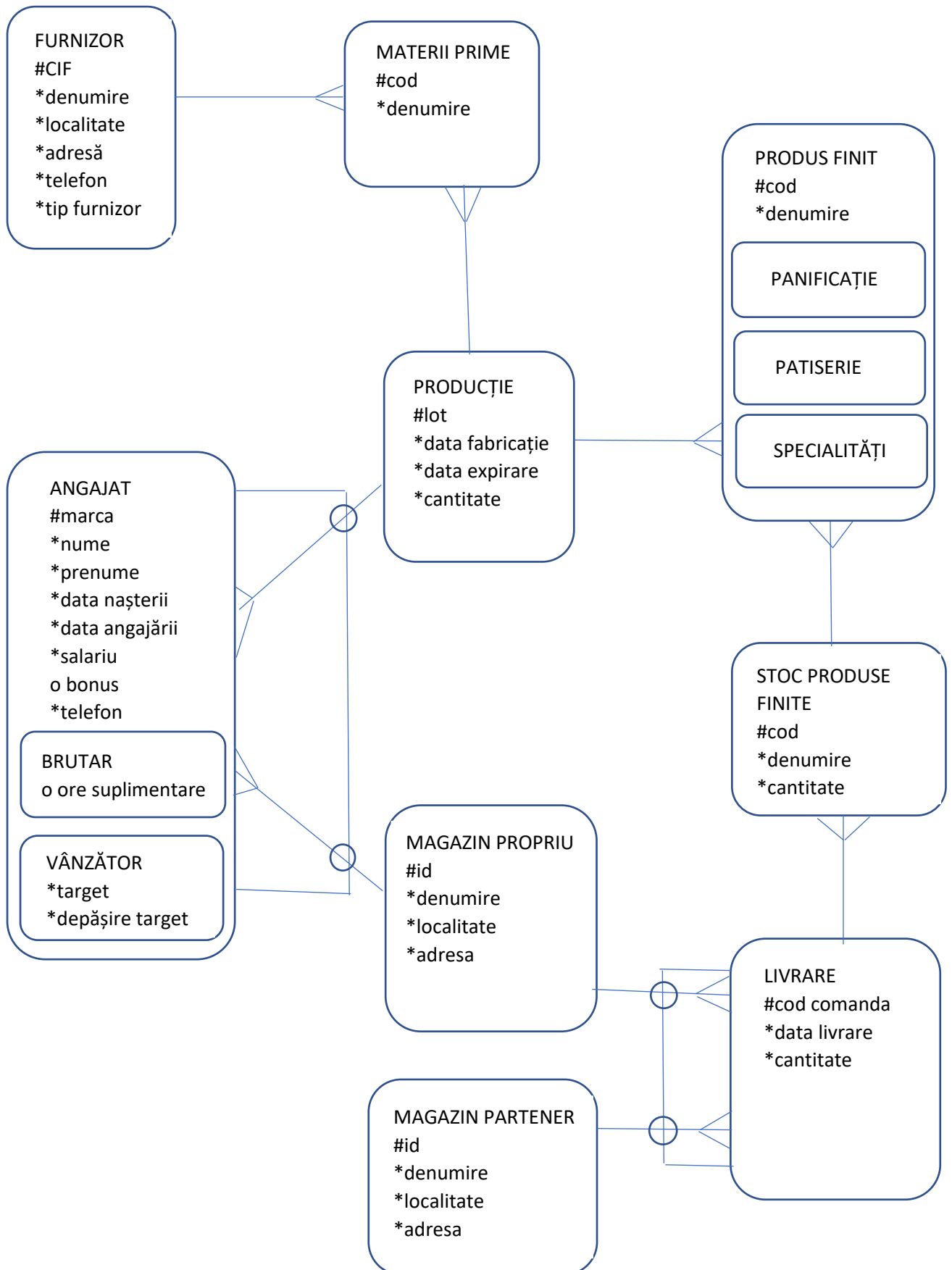
Materiile prime, necesare procesului de producție, provin de la furnizori cu care au fost încheiate contracte în prealabil pentru livrare de făină, ouă, lapte, sare, drojdie, etc. Fiecare furnizor livrează un singur tip de materie primă de exemplu societatea Olympus SRL livrează dulceață care poate fi de afine, zmeură etc.

Angajații societății comerciale își pot desfășura activitatea în departamentul de producție alcatuit dintr-o echipă de brutari cu experiență, bine pregătiți sau în departamentul de vânzări. Toți angajații beneficiază de contracte de muncă, cu program de 8 ore.

Fiecărui angajat îi este alocat un identificator unic, sub formă de marcă, pentru a se diferenția. De asemenea, în baza de date, tabela angajați va conține informații referitoare la numele acestora, data nașterii, data angajării, salariu, număr de telefon și departamentul din care aceștia fac parte. Pentru a face față comenzilor, în anumite perioade mai aglomerate, cum ar fi cea a sărbătorilor, brutarii pot lucra ore suplimentare plătite în plus. De asemenea, vânzătorii, pe lângă salariu pot primi bonusuri comerciale din vânzări în cazul depășirii target-ului, reprezentând vânzări de peste 50 000 RON/lună pentru fiecare angajat în parte.

Pe lângă magazinele proprii, firma a încheiat contracte de colaborare cu mariile lanțuri de magazine cum ar fi Profi, Kaufland, Carrefour pentru comercializarea produselor. Livrarea produselor se realizează prin colaborare cu o firmă specializată în transportul acestora.

## Diagrama ERD



## Crear tabelle

[illegible]

```
cod_lvr NUMBER(6) NOT NULL REFERENCES livrari(cod));
```

```
CREATE TABLE produse_finite(cod NUMBER(4) PRIMARY KEY,  
    denumire VARCHAR2(35) NOT NULL,  
    tip_pft VARCHAR2(15) NOT NULL,  
    lot_pde NUMBER(4) NOT NULL REFERENCES productii(lot),  
    cod_stm NUMBER(6) NOT NULL REFERENCES stoc_marfuri(cod));
```

```
CREATE TABLE productii(lot NUMBER(6) PRIMARY KEY,  
    data_fabricare DATE NOT NULL,  
    data_expirare DATE NOT NULL,  
    cantitate NUMBER(7) NOT NULL);
```

```
CREATE TABLE materii_prime(cod NUMBER(6) PRIMARY KEY,  
    denumire VARCHAR2(25) NOT NULL,  
    CIF_fnz NUMBER(6) NOT NULL REFERENCES furnizori(CIF),  
    lot_pde NUMBER(6) NOT NULL REFERENCES productii(lot));
```

```
CREATE TABLE angajati(marca NUMBER(2) PRIMARY KEY,  
    nume VARCHAR2(20) NOT NULL,  
    prenume VARCHAR2(20) NOT NULL,  
    data_nasteri DATE NOT NULL,  
    data_angajarii DATE NOT NULL,  
    salariu NUMBER(4) NOT NULL,  
    bonus NUMBER(4),  
    telefon NUMBER(10) NOT NULL,  
    ore_suplim NUMBER(2),  
    target NUMBER(8),  
    depasire_tgt NUMBER(8),  
    tip_agt CHAR(1) NOT NULL,  
    lot_pde NUMBER(6) REFERENCES productii(lot),
```

id\_mpr NUMBER(6) REFERENCES magazine\_propri(id));

## Popularea tabelelor

### POPULARE FURNIZORI:

INSERT INTO furnizori VALUES(111111, 'EUROPAN SRL', 'Sighisoara', 'Str. Garoafelor nr.18', 0733726473, 'faina');

INSERT INTO furnizori VALUES(222222, 'FERMA SRL', 'Sovata', 'Str. Mihai Viteazu nr.2', 0733827465, 'oua');

INSERT INTO furnizori VALUES(333333, 'COVALACT SRL', 'Tg. Mures', 'Str. Florilor nr.8', 0733917456, 'lapte');

INSERT INTO furnizori VALUES(444444, 'AGROSER SRL', 'Reghin', 'Zoriilor nr.15', 0744213484, 'seminte');

INSERT INTO furnizori VALUES(555555, 'LAURUS SRL', 'Sighisoara', 'Str. Ana Ipatesu nr.20', 0734619433, 'zahar');

INSERT INTO furnizori VALUES(666666, 'OLYMPUS SRL', 'Sighisoara', 'Str. Hermann Oberth nr.11', 0743885943, 'dulceata');

INSERT INTO furnizori VALUES(777777, 'AGRO SRL', 'Danes', 'Str. Andrei Shaguna nr.20', 0788441122, 'drojdie');

CIF	DENUMIRE	LOCALITATE	ADRESA	TELEFON	TIP_FURNIZOR
111111	EUROPAN SRL	Sighisoara	Str. Garoafelor nr.18	733726473	faina
222222	FERMA SRL	Sovata	Str. Mihai Viteazu nr.2	733827465	oua
333333	COVALACT SRL	Tg. Mures	Str. Florilor nr.8	733917456	lapte
666666	OLYMPUS SRL	Sighisoara	Str. Hermann Oberth nr.11	743885943	dulceata
444444	AGROSER SRL	Reghin	Zoriilor nr.15	744213484	seminte
555555	LAURUS SRL	Sighisoara	Str. Ana Ipatesu nr.20	734619433	zahar
777777	AGRO SRL	Danes	Str. Andrei Shaguna nr.20	788441122	drojdie

### POPULARE MATERII\_PRIME:

INSERT INTO materii\_prime VALUES(100, 'ZAHAR BRUN', 555555, 1122);

INSERT INTO materii\_prime VALUES(101, 'ZAHAR TOS', 555555, 3344);

INSERT INTO materii\_prime VALUES(160, 'FAINA DE GRAU', 111111, 1122);

INSERT INTO materii\_prime VALUES(345, 'SEMINTE DE DOVLEAC', 444444, 5533);

INSERT INTO materii\_prime VALUES(201, 'DULCEATA DE AFINE', 666666, 4422);

INSERT INTO materii\_prime VALUES(555, 'DROJDIE', 777777, 1122);

COD	DENUMIRE	CIF_FNZ	LOT_PDE
101	ZAHAR TOS	555555	3344
130	LAPTE DE VACA	333333	2211
160	FAINA DE GRAU	111111	1122
100	ZAHAR BRUN	555555	1122
345	SEMINTE DE DOVLEAC	444444	5533
201	DULCEATA DE AFINE	666666	4422
202	DULCEATA DE ZMEURA	666666	5566
555	DROJDIE	777777	1122
121	OUA	222222	4455

## POPULARE PRODUCTII:

INSERT INTO productii VALUES(1122, TO\_DATE('05/04/2022','DD,MM,YYYY'),  
TO\_DATE('08/04/2022','DD,MM,YYYY'), 300);

INSERT INTO productii VALUES(3344, TO\_DATE('05/04/2022','DD,MM,YYYY'),  
TO\_DATE('08/04/2022','DD,MM,YYYY'), 500);

INSERT INTO productii VALUES(2211, TO\_DATE('05/04/2022','DD,MM,YYYY'),  
TO\_DATE('08/04/2022','DD,MM,YYYY'), 200);

INSERT INTO productii VALUES(5533, TO\_DATE('05/04/2022','DD,MM,YYYY'),  
TO\_DATE('08/04/2022','DD,MM,YYYY'), 300);

INSERT INTO productii VALUES(4422, TO\_DATE('05/04/2022','DD,MM,YYYY'),  
TO\_DATE('08/04/2022','DD,MM,YYYY'), 400);

INSERT INTO productii VALUES(5566, TO\_DATE('05/04/2022','DD,MM,YYYY'),  
TO\_DATE('08/04/2022','DD,MM,YYYY'), 100);

INSERT INTO productii VALUES(4455, TO\_DATE('05/04/2022','DD,MM,YYYY'),  
TO\_DATE('08/04/2022','DD,MM,YYYY'), 250);

LOT	DATA_FABRICARE	DATA_EXPIRARE	CANTITATE
5566	05-Apr-2022	08-Apr-2022	100
3344	05-Apr-2022	08-Apr-2022	500
4422	05-Apr-2022	08-Apr-2022	400
4455	05-Apr-2022	08-Apr-2022	250
1122	05-Apr-2022	08-Apr-2022	300
2211	05-Apr-2022	08-Apr-2022	200
5533	05-Apr-2022	08-Apr-2022	300

## POPULARE PRODUSE\_FINITE:

INSERT INTO produse\_finite VALUES(1111, 'PAINE ALBA', 'PANIFICATIE', 3344, 1);

```

INSERT INTO produse_finite VALUES(2222, 'COVRIG SIMPLU', 'PATISERIE', 1122, 2);
INSERT INTO produse_finite VALUES(1112, 'PAINE CU SEMINTE', 'PANIFICATIE', 2211, 3);
INSERT INTO produse_finite VALUES(1113, 'PAINE NEAGRA', 'PANIFICATIE', 5533, 7);
INSERT INTO produse_finite VALUES(2223, 'COVRIG CU SUSAN', 'PATISERIE', 4422, 8);
INSERT INTO produse_finite VALUES(2233, 'STRUDEL CU MERE', 'PATISERIE', 5566, 5);
INSERT INTO produse_finite VALUES(2234, 'FOETAJ CU AFINE', 'PATISERIE', 4455, 4);
INSERT INTO produse_finite VALUES(3333, 'COZONAC', 'SPECIALITATE', 5533, 6);

```

COD	DENUMIRE	TIP_PFT	LOT_PDE	COD_STM
1113	PAINE NEAGRA	PANIFICATIE	5533	7
2222	COVRIG SIMPLU	PATISERIE	1122	2
2223	COVRIG CU SUSAN	PATISERIE	4422	8
2233	STRUDEL CU MERE	PATISERIE	5566	5
2234	FOETAJ CU AFINE	PATISERIE	4455	4
3333	COZONAC	SPECIALITATE	5533	6
1111	PAINE ALBA	PANIFICATIE	3344	1
1112	PAINE CU SEMINTE	PANIFICATIE	2211	3

#### POPULARE STOC\_MARFURI:

```

INSERT INTO stoc_produse_finite VALUES(1, 'PAINE ALBA', 10000, 102);
INSERT INTO stoc_produse_finite VALUES(2, 'COVRIG SIMPLU', 8000, 102);
INSERT INTO stoc_produse_finite VALUES(3, 'PAINE CU SEMINTE', 10000, 103);
INSERT INTO stoc_produse_finite VALUES(4, 'FOETAJ CU AFINE', 5000, 104);
INSERT INTO stoc_produse_finite VALUES(5, 'STRUDEL CU MERE', 8000, 105);
INSERT INTO stoc_produse_finite VALUES(6, 'COZONAC', 5000, 101);
INSERT INTO stoc_produse_finite VALUES(7, 'PAINE NEAGRA', 5000, 105);
INSERT INTO stoc_produse_finite VALUES(8, 'COVRIG CU SUSAN', 5000, 106);

```



COD	DENUMIRE	CANTITATE	COD_LVR
1	PAINE ALBA	10000	102
2	COVRIG SIMPLU	9000	102
3	PAINE CU SEMINTE	10000	103
4	FOETAJ CU AFINE	8000	104
5	STRUDEL CU MERE	8000	105
6	COZONAC	5000	101
7	PAINE NEAGRA	5000	105
8	COVRIG CU SUSAN	5000	106

#### POPULARE LIVRARI:

INSERT INTO livrari VALUES(101, TO\_DATE('05/04/2022','DD,MM,YYYY'), 3000, 12, NULL);

INSERT INTO livrari VALUES(102, TO\_DATE('06/04/2022','DD,MM,YYYY'), 5000, 13, NULL);

INSERT INTO livrari VALUES(103, TO\_DATE('06/04/2022','DD,MM,YYYY'), 5000, 14, NULL);

INSERT INTO livrari VALUES(104, TO\_DATE('05/04/2022','DD,MM,YYYY'), 6000, NULL, 22);

INSERT INTO livrari VALUES(105, TO\_DATE('05/04/2022','DD,MM,YYYY'), 6000, NULL, 33);

INSERT INTO livrari VALUES(106, TO\_DATE('05/04/2022','DD,MM,YYYY'), 6000, NULL, 44);

COD	DATA_LIVRARE	CANTITATE	ID_MGP	ID_MPT
102	06-Apr-2022	5000	13	-
103	06-Apr-2022	5000	14	-
104	05-Apr-2022	6000	-	22
105	05-Apr-2022	6000	-	33
106	05-Apr-2022	6000	-	44
101	05-Apr-2022	3000	12	-

#### POPULARE MAGAZINE\_PROPRII:

INSERT INTO magazine\_propri VALUES(12, 'Eldi', 'Sighisoara', 'Str. Lalelelor nr.23');

INSERT INTO magazine\_propri VALUES(13, 'Eldi', 'Sighisoara', 'Str. Nicolae Titulescu nr.34');

INSERT INTO magazine\_propri VALUES(14, 'Eldi', 'Tg. Mures', 'Str. Gheorghe Doja nr.21');

INSERT INTO magazine\_propri VALUES(15, 'Eldi', 'Tg. Mures', 'Str. Libertati nr.45');

INSERT INTO magazine\_propri VALUES(16, 'Eldi', 'Sovata', 'Str. Andrei Saguna nr.35');

INSERT INTO magazine\_propri VALUES(17, 'Eldi', 'Sovata', 'Str. Crizantemelor nr.11');

ID	DENUMIRE	LOCALITATE	ADRESA
12	Eldi	Sighisoara	Str. Lalelelor nr.23
13	Eldi	Sighisoara	Str. Nicolae Titulescu nr.34
14	Eldi	Sighisoara	Str. Gheorghe Doja nr.21
15	Eldi	Tg. Mures	Str. Libertati nr.45
16	Eldi	Sovata	Str. Andrei Saguna nr.35
17	Eldi	Sovata	Str. Crizantemelor nr.11

#### POPULARE MAGAZINE\_PARTENERE:

INSERT INTO magazine\_partenere VALUES(22, 'Profi', 'Sighisoara', 'Str. Hermann Oberth nr.10');

INSERT INTO magazine\_partenere VALUES(33, 'Kaufland', 'Sighisoara', 'Str. Crizantemelor nr.23');

INSERT INTO magazine\_partenere VALUES(44, 'Carrefour', 'Tg.Mures', 'Str. Aurel Mosora, nr. 11');

INSERT INTO magazine\_partenere VALUES(55, 'Profi', 'Sighisoara', 'Str. Trandafirilor nr. 12');

INSERT INTO magazine\_partenere VALUES(66, 'Kaufland', 'Tg. Mures', 'Str. Alexandru Vaida Voevod nr.20');

INSERT INTO magazine\_partenere VALUES(77, 'Profi', 'Sovata', 'Str. Aurel Vlaicu nr.14');

ID	DENUMIRE	LOCALITATE	ADRESA
77	Profi	Sovata	Str. Aurel Vlaicu nr.14
22	Profi	Sighisoara	Str. Hermann Oberth nr.10
33	Kaufland	Sighisoara	Str. Crizantemelor nr.23
44	Carrefour	Tg.Mures	Str. Aurel Mosora, nr. 11
55	Profi	Sighisoara	Str. Trandafirilor nr. 12
66	Kaufland	Tg. Mures	Str. Alexandru Vaida Voevod nr.20

#### POPULARE ANGAJAȚI:

INSERT INTO angajati VALUES(1, 'Popescu', 'Ion', TO\_DATE('27/04/1995','DD,MM,YYYY'),  
TO\_DATE('01/02/2020','DD,MM,YYYY'), 2500, NULL, 0733224592, NULL, NULL, NULL, 'B', 1122,  
NULL);

INSERT INTO angajati VALUES(2, 'Babos', 'Ilinca', TO\_DATE('13/02/1990','DD,MM,YYYY'),  
TO\_DATE('10/04/2010','DD,MM,YYYY'), 2500, 100, 0733224592, NULL, 50000, 2000 , 'V', NULL, 12);

INSERT INTO angajati VALUES(3, 'Veres', 'Cristina', TO\_DATE('01/08/1999','DD,MM,YYYY'),  
TO\_DATE('20/05/2021','DD,MM,YYYY'), 2500, NULL, 0733224592, NULL, NULL, NULL, 'V', NULL, 13);

INSERT INTO angajati VALUES(1, 'Ionescu', 'Mihai', TO\_DATE('28/05/1993','DD,MM,YYYY'),  
TO\_DATE('21/04/2015','DD,MM,YYYY'), 2500, 100, 0733224592, 20, NULL, NULL, 'B', 1122, NULL);

```
INSERT INTO angajati VALUES(1, 'Man', 'Rebeca', TO_DATE('21/03/1995','DD,MM,YYYY'),
TO_DATE('01/01/2015','DD,MM,YYYY'), 2500, NULL, 0733224592, NULL, NULL, NULL, 'B', 3344,
NULL);
```

```
INSERT INTO angajati VALUES(1, 'Marcu', 'Ionut', TO_DATE('05/04/1998','DD,MM,YYYY'),
TO_DATE('03/02/2019','DD,MM,YYYY'), 2500, 100, 0733224592, NULL, 50000, 2000, 'V', NULL, 14);
```

MARCA	NUME	PRENUME	DATA_NASTERI	DATA_ANGAJARII	SALARIU	BONUS	TELEFON	ORE_SUPLIM	TARGET	DEPASIRE_TGT	TIP_AGT	LOT_PDE	ID_MPR
2	Babos	Ilinca	13-Feb-1990	10-Apr-2010	2500	100	733224592	-	50000	2000	V	-	12
4	Ionescu	Mihai	28-May-1993	21-Apr-2015	2500	100	733224592	20	-	-	B	1122	-
1	Popescu	Ion	27-Apr-1995	01-Feb-2020	2500	-	733224592	-	-	-	B	1122	-
3	Veres	Cristina	01-Aug-1999	20-May-2021	2500	-	733224592	-	-	-	V	-	13
5	Man	Rebeca	21-Mar-1995	01-Jan-2015	2500	-	733224592	-	-	-	B	3344	-
6	Marcu	Ionut	05-Apr-1998	03-Feb-2019	2500	100	733224592	-	50000	2000	V	-	14

# Creare package specification

CREATE OR REPLACE PACKAGE package\_brutarie IS

marire\_salariu NUMBER := 200; 1. Variabilă publică

data\_curenta DATE := SYSDATE; 2. Variabilă publică

1. Funcție care returnează numărul de angajați

FUNCTION numar\_angajati RETURN NUMBER;

2. Funcție care calculează salariul unui angajat adaugând bonusul acestora

FUNCTION calcul\_salariu(marca\_angajat NUMBER) RETURN NUMBER;

3. Funcție care returnează postul ocupat de un angajat, 'B' pentru Brutar și 'V' pentru Vânzator

FUNCTION tip\_angajat(marca\_angajat NUMBER) RETURN VARCHAR2;

1. Procedură care mărește salariul tuturor angajaților cu 200 RON folosind variabila publică marire\_salariu

PROCEDURE mareste\_salariu;

2. Procedură care șterge un anumit produs din tabelul PRODUSE\_FINITE și afișează produsele rămase

PROCEDURE sterge\_produs(v\_cod NUMBER);

3. Procedură care introduce un angajat în tabelul ANGAJATI

PROCEDURE adauga\_angajat(v\_marca NUMBER, v\_nume VARCHAR2, v\_prenume VARCHAR2, v\_data\_nasteri DATE, v\_data\_angajarii DATE, v\_salariu NUMBER, v\_bonus NUMBER, v\_telefon NUMBER, v\_ore\_suplim NUMBER, v\_target NUMBER, v\_depasire\_tgt NUMBER, v\_tip\_agt CHAR, v\_lot\_pde NUMBER, v\_id\_mpr NUMBER);

4. Procedură care afișează denumirea și cantitatea pentru produsele care au cea mai mare cantitate pe stoc

PROCEDURE afisare\_stoc\_maxim;

5. Procedură care afișează vechimea unui angajat în ani folosind variabila publică data\_curenta

PROCEDURE afisare\_vechime(v\_marca NUMBER);

6. Procedură care afișează toate magazinele proprii și angajați care lucrează în ele

PROCEDURE afisare\_mgp\_angajati;

END package\_brutarie;

## Creare package body

CREATE OR REPLACE PACKAGE BODY package\_brutarie IS

1. [Cursor explicit fără parametri pentru tabelul ANGAJAȚI](#)

CURSOR cursor\_angajati IS

SELECT marca, nume, prenume, data\_nasteri, data\_angajarii, salariu, bonus, telefon, ore\_suplim,  
target, depasire\_tgt, tip\_agt, lot\_pde, id\_mpr

FROM angajati

ORDER BY nume;

2. [Cursor explicit cu parametri pentru tabelul STOC\\_PRODUSE\\_FINITE](#)

CURSOR cursor\_produce\_stoc\_max(cantitate\_max NUMBER) IS

SELECT cod, denumire, cantitate, cod\_lvr

FROM stoc\_produce\_finite

WHERE cantitate = cantitate\_max;

1. [Funcție privată care returnează cea mai mare cantitate dintr-un produs aflat pe stoc](#)

FUNCTION cantitate\_maxima RETURN NUMBER

IS

cantitate\_max NUMBER;

BEGIN

SELECT MAX(cantitate) INTO cantitate\_max FROM stoc\_produce\_finite;

RETURN cantitate\_max;

END cantitate\_maxima;

1. [Procedură privată care afișează produsele din tabelul PRODUSE\\_FINITE](#)

PROCEDURE afisare\_produce IS

CURSOR cursor\_produce IS

SELECT cod, denumire, tip\_pft, lot\_pde, cod\_stm

FROM produce\_finite;

v\_produce\_rec cursor\_produce%ROWTYPE;

```

BEGIN

OPEN cursor_produce;

LOOP

    FETCH cursor_produce INTO v_produce_rec;

    EXIT WHEN cursor_produce%NOTFOUND;

    DBMS_OUTPUT.PUT_LINE('Cod: ' || v_produce_rec.cod || ', Denumire: ' ||
v_produce_rec.denumire ||

    ', Tip produs: ' || v_produce_rec.tip_pft || ', Lot produs: ' || v_produce_rec.lot_pde ||

    ', Cod stoc produse: ' || v_produce_rec.cod_stm);

    DBMS_OUTPUT.PUT_LINE(' ');

END LOOP;

CLOSE cursor_produce;

END afisare_produce;

```

## 2. Procedură privată care incrementează o valoare

```

PROCEDURE increment(i IN OUT NUMBER) IS

BEGIN

    i := i + 1;

END increment;

```

## 1. Funcție care returnează numărul de angajați folosind procedura privată increment(i)

```

FUNCTION numar_angajati RETURN NUMBER

IS

    nr_angajati NUMBER := 0;

    v_angajati_rec cursor_angajati%ROWTYPE;

BEGIN

    OPEN cursor_angajati;

    LOOP

        FETCH cursor_angajati INTO v_angajati_rec;

        EXIT WHEN cursor_angajati%NOTFOUND;

        increment(nr_angajati);
    
```

```

END LOOP;

CLOSE cursor_angajati;

RETURN nr_angajati;

END numar_angajati;

```

## 2. Funcție care calculează salariul unui angajat adăugând bonusul acestora

```

FUNCTION calcul_salariu(marca_angajat NUMBER) RETURN NUMBER
IS
    salariu_total NUMBER := 0;
    angajat angajati%ROWTYPE;
BEGIN
    SELECT * INTO angajat
    FROM angajati
    WHERE marca = marca_angajat;
    IF angajat.bonus != 0 THEN
        salariu_total := angajat.salariu + angajat.bonus;
        DBMS_OUTPUT.PUT_LINE('Angajatul are un bonus de: ' || angajat.bonus);
    ELSE
        salariu_total := angajat.salariu;
    END IF;
    RETURN salariu_total;
END calcul_salariu;

```

## 3. Funcție care returnează postul ocupat de un angajat, 'B' pentru Brutar și 'V' pentru Vânzător

```

FUNCTION tip_angajat(marca_angajat NUMBER) RETURN VARCHAR2
IS
    angajat angajati%ROWTYPE;
BEGIN
    SELECT * INTO angajat
    FROM angajati
    WHERE marca = marca_angajat;

```

```
RETURN angajat.tip_agt;
```

```
END tip_angajat;
```

1. Procedură care mărește salariul tuturor angajaților cu 200 RON folosind variabila publică `mărire_salariu`

```
PROCEDURE mareste_salariu IS
```

```
    v_marire NUMBER := 0;
```

```
    v_angajati_rec cursor_angajati%ROWTYPE;
```

```
BEGIN
```

```
    OPEN cursor_angajati;
```

```
    LOOP
```

```
        FETCH cursor_angajati INTO v_angajati_rec;
```

```
        EXIT WHEN cursor_angajati%NOTFOUND;
```

```
        v_marire := v_angajati_rec.salariu + marire_salariu;
```

```
        UPDATE angajati
```

```
            SET salariu = v_marire;
```

```
    END LOOP;
```

```
    CLOSE cursor_angajati;
```

```
END mareste_salariu;
```

2. Procedură care șterge un anumit produs din tabelul `PRODUSE_FINITE` și afișează produsele rămase folosind procedura privată `afisare_produce`

```
PROCEDURE sterge_produs(v_cod NUMBER) IS
```

```
BEGIN
```

```
    DELETE FROM produse_finite WHERE cod = v_cod;
```

```
    DBMS_OUTPUT.PUT_LINE('Produs sters cu succes!');
```

```
    afisare_produce;
```

```
END;
```

3. Procedură care introduce un angajat în tabelul `ANGAJATI`

```
PROCEDURE adauga_angajat(v_marca NUMBER, v_nume VARCHAR2, v_prenume VARCHAR2,
```

```
v_data_nasteri DATE, v_data_angajarii DATE, v_salariu NUMBER, v_bonus NUMBER, v_telefon
```



```
NUMBER, v_ore_suplim NUMBER, v_target NUMBER, v_depasire_tgt NUMBER, v_tip_agt CHAR,  
v_lot_pde NUMBER, v_id_mpr NUMBER)
```

```
IS
```

```
BEGIN
```

```
    INSERT INTO angajati(marca, nume, prenume, data_nasteri, data_angajarii, salariu, bonus,  
    telefon, ore_suplim, target, depasire_tgt, tip_agt, lot_pde, id_mpr)
```

```
    VALUES (v_marca, v_nume, v_prenume, v_data_nasteri, v_data_angajarii, v_salariu, v_bonus,  
    v_telefon, v_ore_suplim, v_target, v_depasire_tgt, v_tip_agt, v_lot_pde, v_id_mpr);
```

```
    DBMS_OUTPUT.PUT_LINE('Angajat adaugat cu succes!');
```

```
END adauga_angajat;
```

#### 4. Procedură care afișează denumirea și cantitatea pentru produsele care au cea mai mare cantitate pe stoc folosind funcția privată `cantitate_maximă`

```
PROCEDURE afisare_stoc_maxim IS
```

```
v_cantitate_max NUMBER := cantitate_maxima;
```

```
v_produce_stoc_rec cursor_produce_stoc_max%ROWTYPE;
```

```
BEGIN
```

```
    OPEN cursor_produce_stoc_max(v_cantitate_max);
```

```
    LOOP
```

```
        FETCH cursor_produce_stoc_max INTO v_produce_stoc_rec;
```

```
        EXIT WHEN cursor_produce_stoc_max%NOTFOUND;
```

```
        DBMS_OUTPUT.PUT_LINE('Denumire: ' || v_produce_stoc_rec.denumire || ', cantitate: ' ||
```

```
        v_produce_stoc_rec.cantitate);
```

```
    END LOOP;
```

```
    CLOSE cursor_produce_stoc_max;
```

```
END afisare_stoc_maxim;
```

#### 5. Procedură care afișează vechimea unui angajat în ani folosind variabila publică `data_curenta`

```
PROCEDURE afisare_vechime(v_marca NUMBER) IS
```

```
v_ang angajati%ROWTYPE;
```

```

v_vechime NUMBER;

BEGIN

    SELECT nume, prenume, data_angajarii INTO v_ang.nume, v_ang.prenume,
v_ang.data_angajarii
    FROM angajati
    WHERE marca = v_marca;

    v_vechime := TRUNC(MONTHS_BETWEEN(data_curenta, v_ang.data_angajarii)/12);

    DBMS_OUTPUT.PUT_LINE('Angajatul cu numele: ' || v_ang.nume || ' ' || v_ang.prenume);
    DBMS_OUTPUT.PUT_LINE('Data angajari: ' || v_ang.data_angajarii);
    DBMS_OUTPUT.PUT_LINE('Vechime: ' || v_vechime || ' ani');

END;

```

#### 6. Procedură care afișează toate magazinele propri și angajați care lucrează în ele folosind un cursor multiplu

```

PROCEDURE afisare_mgp_angajati IS

CURSOR cursor_mgp IS

    SELECT id, denumire, localitate, adresa
    FROM magazine_propri;

CURSOR cursor_angajati_vanzatori(p_id_mgp NUMBER) IS

    SELECT nume, prenume, id_mpr
    FROM angajati
    WHERE tip_agt='V' AND id_mpr = p_id_mgp
    ORDER BY nume;

v_mgp_rec cursor_mgp%ROWTYPE;
v_angajati_vanzatori_rec cursor_angajati_vanzatori%ROWTYPE;

BEGIN

    OPEN cursor_mgp;

    LOOP

        FETCH cursor_mgp INTO v_mgp_rec;

        EXIT WHEN cursor_mgp%NOTFOUND;

        DBMS_OUTPUT.PUT_LINE('In magazinul: ' || v_mgp_rec.denumire || ' din localitatea: ' ||
v_mgp_rec.localitate || ', adresa: ' || v_mgp_rec.adresa || ' lucreaza: ');
    
```

```

OPEN cursor_angajati_vanzatori(v_mgp_rec.id);

LOOP

    FETCH cursor_angajati_vanzatori INTO v_angajati_vanzatori_rec;

    EXIT WHEN cursor_angajati_vanzatori%NOTFOUND;

    DBMS_OUTPUT.PUT_LINE('Angajatul: ' || v_angajati_vanzatori_rec.ume || ' ' ||
        v_angajati_vanzatori_rec.prenume);

    DBMS_OUTPUT.PUT_LINE(' ');

END LOOP;

CLOSE cursor_angajati_vanzatori;

END LOOP;

CLOSE cursor_mgp;

END afisare_mgp_angajati;

END package_brutarie;

```

## Creare declanșatori (triggers)

1. [Trigger pentru afișarea venitului vechi, venitului nou și a diferenței dintre cele două venituri după ce se mărește salariul angajaților](#)

```

CREATE OR REPLACE TRIGGER marire_salariu_angajati
    BEFORE UPDATE OF salariu ON angajati FOR EACH ROW
DECLARE
    diferenta NUMBER := 0;
BEGIN
    diferenta := :NEW.salariu - :OLD.salariu;

    DBMS_OUTPUT.PUT_LINE('Salariul anterior: ' || :OLD.salariu);
    DBMS_OUTPUT.PUT_LINE('Salariul actual: ' || :NEW.salariu);
    DBMS_OUTPUT.PUT_LINE('Diferenta dintre cele doua salarii: ' || diferenta || ' RON');
END;

```

Results

Explain

Describe

Saved SQL

History

Trigger created.

0.02 seconds

## 2. Trigger care nu permite inserarea unui angajat cu un salariu mai mic de 2500 RON

```
CREATE OR REPLACE TRIGGER adauga_angajat
```

```
BEFORE INSERT ON angajati FOR EACH ROW
```

```
BEGIN
```

```
IF :NEW.salariu < 2500 THEN
```

```
    RAISE_APPLICATION_ERROR(-20001, 'Angajati nu pot avea un salariu mai mic de 2500 RON');
```

```
END IF;
```

```
END;
```

## 3. Trigger care, după ștergerea unui produs, stochează datele acestuia într-un tabel

```
CREATE TABLE log_produce(data VARCHAR2(20),
```

```
    denumire VARCHAR(50),
```

```
    tip VARCHAR2(20));
```

```
CREATE OR REPLACE TRIGGER sterge_produc
```

```
AFTER DELETE ON produse_finite FOR EACH ROW
```

```
BEGIN
```

```
    INSERT INTO log_produce (data, denumire, tip)
```

```
    VALUES (SYSDATE, :OLD.denumire, :OLD.tip_pft);
```

```
END;
```

## 4. Trigger care, după ștergerea unui angajat, stochează datele acestuia într-un tabel

```
CREATE TABLE log_angajati(data VARCHAR2(20),
```

```
    marca NUMBER(8),
```

```
    nume VARCHAR(30),
```

```
    prenume VARCHAR2(30),
```

```
data_angajarii DATE,  
salariu NUMBER(8),  
tip_angajat CHAR(1));
```

```
CREATE OR REPLACE TRIGGER sterge_angajat  
AFTER DELETE ON angajati FOR EACH ROW  
BEGIN  
    INSERT INTO log_angajati (data, marca, nume, prenume, data_angajarii, salariu, tip_angajat)  
    VALUES (SYSDATE, :OLD.marca, :OLD.nume, :OLD.prenume, :OLD.data_angajarii, :OLD.salariu,  
:OLD.tip_agt);  
END;
```

## Creare secvențe

```
CREATE SEQUENCE secventa_magazine_partenere START WITH 88 INCREMENT BY 11  
CREATE SEQUENCE secventa_angajati START WITH 9 INCREMENT BY 1;
```

Results	Explain	Describe	Saved SQL	History
---------	---------	----------	-----------	---------

Sequence created.

## Testarea funcțiilor și a procedurilor

1. Funcție care returnează numărul de angajați folosind funcția privată increment(i)

```
BEGIN  
    DBMS_OUTPUT.PUT_LINE('In Brutarie sunt: ' || package_brutarie.numar_angajati || ' angajati');  
END;
```

Results	Explain	Describe	Saved SQL	History
---------	---------	----------	-----------	---------

In Brutarie sunt: 7 angajati

Statement processed.

2. Funcție care calculează salariul unui angajat adăugând bonusul acestora

BEGIN

```
DBMS_OUTPUT.PUT_LINE('Salariul este: ' || package_brutarie.calcul_salariu(4));
```

END;

Results	Explain	Describe	Saved SQL	History
Angajatul are un bonus de: 100 Salariul este: 3000				

3. Funcție care returnează postul ocupat de un angajat, 'B' pentru Brutar și 'V' pentru Vânzator

BEGIN

```
DBMS_OUTPUT.PUT_LINE('Angajatul este: ' || package_brutarie.tip_angajat(4));
```

END;

Results	Explain	Describe	Saved SQL	History
Angajatul este: B  Statement processed.				

1. Procedură care mărește salariul tuturor angajaților cu 200 RON folosind variabila publică `mărire_salariu`

```
SELECT nume, prenume, salariu FROM angajati;
```

Results	Explain	Describe	Saved SQL	History
NUME			PRENUME	
			SALARIU	
Babos			Ilincă	
Ionescu			Mihai	
Pop			Maria	
Popescu			Ion	
Veres			Cristina	
Man			Rebeca	
Marcu			Ionut	

```
package_brutarie.mareste_salariu;  
END;
```

2. Procedură care șterge un anumit produs din tabelul PRODUSE\_FINITE și afișează produsele rămase folosind procedura privată afisare\_produce

Results	Explain	Describe	Saved SQL	History
COD	DENUMIRE	TIP_PFT	LOT_PDE	COD_STM
4444	STRUDEL CU VISINE	PATISERIE	5533	6
1113	PAINE NEAGRA	PANIFICATIE	5533	7
2222	COVRIG SIMPLU	PATISERIE	1122	2
2223	COVRIG CU SUSAN	PATISERIE	4422	8
2233	STRUDEL CU MERE	PATISERIE	5566	5

```
package_brutarie.sterge_produs(4444);  
END;
```

Results	Explain	Describe	Saved SQL	History
---------	---------	----------	-----------	---------

Produs sters cu succes!

Cod: 1113, Denumire: PAINE NEAGRA, Tip produs: PANIFICATIE, Lot produs: 5533, Cod stoc produse: 7

Cod: 2222, Denumire: COVRIG SIMPLU, Tip produs: PATISERIE, Lot produs: 1122, Cod stoc produse: 2

Cod: 2223, Denumire: COVRIG CU SUSAN, Tip produs: PATISERIE, Lot produs: 4422, Cod stoc produse: 8

Cod: 2233, Denumire: STRUDEL CU MERE, Tip produs: PATISERIE, Lot produs: 5566, Cod stoc produse: 5

Cod: 2234, Denumire: FOETAJ CU AFINE, Tip produs: PATISERIE, Lot produs: 4455, Cod stoc produse: 4

Cod: 3333, Denumire: COZONAC, Tip produs: SPECIALITATE, Lot produs: 5533, Cod stoc produse: 6

### 3. Procedură care introduce un angajat în tabelul ANGAJATI

BEGIN

```
package_brutarie.adauga_angajat(8, 'Marcu', 'Cristina', '12-Apr-1998', '01-Mar-2022', 3000, NULL,
0177281625, NULL, 40000, NULL, 'V', NULL, 13);
```

END;

Results	Explain	Describe	Saved SQL	History
---------	---------	----------	-----------	---------

MARCA	NUME	PRENUME	DATA_NASTERI	DATA_ANGAJARII	SALARIU	BONUS	TELEFON
2	Babos	Ilinca	13-Feb-1990	10-Apr-2010	2900	100	733224592
7	Pop	Maria	12-Apr-1999	01-Feb-2022	2900	-	182736251
8	Marcu	Cristina	12-Apr-1998	01-Mar-2022	3000	-	177281625

### 4. Procedură care afișează denumirea și cantitatea pentru produsele care au cea mai mare cantitate pe stoc folosind funcția privată cantitate\_maximă

BEGIN

```
package_brutarie.afisare_stoc_maxim;
```

END;

Results	Explain	Describe	Saved SQL	History
---------	---------	----------	-----------	---------

Denumire: PAINE ALBA, cantitate: 10000

Denumire: PAINE CU SEMINTE, cantitate: 10000



5. Procedură care afișează vechimea unui angajat în ani folosind variabila publică data\_curenta

BEGIN

```
package_brutarie.afisare_vechime(4);
```

END;

Results	Explain	Describe	Saved SQL	History
Angajatul cu numele: Ionescu Mihai Data angajari: 21-Apr-2015 Vechime: 7 ani				

6. Procedură care afișează toate magazinele propri și angajați care lucrează în ele folosind un cursor multiplu

BEGIN

```
package_brutarie.afisare_mgp_angajati;
```

END;

Results	Explain	Describe	Saved SQL	History
In magazinul: Eldi din localitatea: Sighisoara, adresa: Str. Lalelelor nr.23 lucreaza: Angajatul: Babos Ilinca  Angajatul: Pop Maria  In magazinul: Eldi din localitatea: Sighisoara, adresa: Str. Nicolae Titulescu nr.34 lucreaza: Angajatul: Marcu Cristina  Angajatul: Veres Cristina  In magazinul: Eldi din localitatea: Sighisoara, adresa: Str. Gheorghe Doja nr.21 lucreaza: Angajatul: Marcu Ionut				

## Testare declanșatori (triggers)

1. Trigger pentru afișarea venitului vechi, venitului nou și a diferenței dintre cele două venituri după ce se mărește salariul angajaților

BEGIN

```
package_brutarie.mareste_salariu;
```

END;

Results	Explain	Describe	Saved SQL	History
Salariul anterior: 3100 Salariul actual: 3300 Diferenta dintre cele doua salarii: 200 RON Salariul anterior: 3100 Salariul actual: 3300 Diferenta dintre cele doua salarii: 200 RON Salariul anterior: 3100 Salariul actual: 3300 Diferenta dintre cele doua salarii: 200 RON				

## 2. Trigger care nu permite inserarea unui angajat cu un salariu mai mic de 2500 RON

BEGIN

```
package_brutarie.adauga_angajat(9, 'Ionescu', 'Laura', '12-Apr-1999', '01-Feb-2022', 2000, NULL,
0182736251, NULL, 40000, NULL, 'V', NULL, 12);
```

END;

Results	Explain	Describe	Saved SQL	History
ORA-20001: Angajati nu pot avea un salariu mai mic de 2500 RON ORA-06512: at "RO_A936_SQL_S32.ADAUGA_ANGAJAT", line 3 ORA-04088: error during execution of trigger 'RO_A936_SQL_S32.ADAUGA_ANGAJAT'  1. BEGIN 2.   package_brutarie.adauga_angajat(9, 'Ionescu', 'Laura', '12-Apr-1999', '01-Feb-2022', 2000, NULL, 3.    0182736251, NULL, 40000, NULL, 'V', NULL, 12); 4. END;				

## 3. Trigger care, după ștergerea unui produs, stochează datele acestuia într-un tabel

BEGIN

```
package_brutarie.sterge_produș(4444);
```

END;

```
SELECT * FROM log_produșe;
```

Results	Explain	Describe	Saved SQL	History
DATA	COD	DENUMIRE	TIP	
02-Jan-2023	4444	STRUDEL CU VISINE	PATISERIE	

4. Trigger care, după ștergerea unui angajat, stochează datele acestuia într-un tabel

```
BEGIN
```

```
DELETE FROM angajati WHERE marca = 9;
```

```
END;
```

Results	Explain	Describe	Saved SQL	History
---------	---------	----------	-----------	---------

```
1 row(s) deleted.
```

```
SELECT * FROM log_angajati;
```

Results	Explain	Describe	Saved SQL	History
---------	---------	----------	-----------	---------

DATA	MARCA	NUME	PRENUME	DATA_ANGAJARII	SALARIU	TIP_ANGAJAT
02-Jan-2023	9	Ionescu	Laura	01-Feb-2022	3000	V

## Testare secvențe

1. Secvența 1

```
CREATE OR REPLACE PROCEDURE adauga_magazine_partenere IS - Bloc cu nume
```

```
BEGIN
```

```
INSERT INTO magazine_partenere VALUES(secventa_magazine_partenere.NEXTVAL, 'Profi',
```

```
'Tg. Mures', 'Str. Andrei Saguna nr.14');
```

```
END;
```

Results	Explain	Describe	Saved SQL	History
---------	---------	----------	-----------	---------

```
Procedure created.
```

```
BEGIN
```

```
adauga_magazine_partenere;
```

```
END;
```

Results	Explain	Describe	Saved SQL	History
---------	---------	----------	-----------	---------

```
Statement processed.
```

SELECT \* FROM magazine\_partenere;

Results	Explain	Describe	Saved SQL	History
ID	DENUMIRE	LOCALITATE	ADRESA	
77	Profi	Sovata	Str. Aurel Vlaicu nr.14	
22	Profi	Sighisoara	Str. Hermann Oberth nr.10	
33	Kaufland	Sighisoara	Str. Crizantemelor nr.23	
44	Carrefour	Tg.Mures	Str. Aurel Mosora, nr. 11	
55	Profi	Sighisoara	Str. Trandafirilor nr. 12	
66	Kaufland	Tg. Mures	Str. Alexandru Vaida Voevod nr.20	
88	Profi	Tg. Mures	Str. Andrei Saguna nr.14	

## 2. Secvența 2

CREATE OR REPLACE PROCEDURE adauga\_angajat\_secventa IS - Bloc cu nume

BEGIN

INSERT INTO angajati VALUES(secventa\_angajati.NEXTVAL, 'Ionescu', 'Laura', '12-Apr-1999', '01-Feb-2022', 3000, NULL, 0182736251, NULL, 40000, NULL, 'V', NULL, 12);

END;

Results	Explain	Describe	Saved SQL	History
---------	---------	----------	-----------	---------

Procedure created.

BEGIN

adauga\_angajat\_secventa;

END;

Results	Explain	Describe	Saved SQL	History
---------	---------	----------	-----------	---------

Statement processed.

SELECT \* FROM angajati WHERE marca = 9;

Results									
Explain   Describe   Saved SQL   History									
MARCA	NUME	PRENUME	DATA_NASTERI	DATA_ANGAJARII	SALARIU	BONUS	TELEFON	ORE_SUPLIM	TARGET
9	Ionescu	Laura	12-Apr-1999	01-Feb-2022	3000	-	182736251	-	40000