

**Academia de Studii Economice București**

**Facultatea de Cibernetică, Statistică și Informatică Economică**

**PROIECT**

**SGBD ORACLE**

Gestiunea unui magazin de produse cosmetice

Profesor coordonator: Un proiect realizat de:

Conf. univ. dr. DIACONIȚA Vlad Țone Iulia-Paula

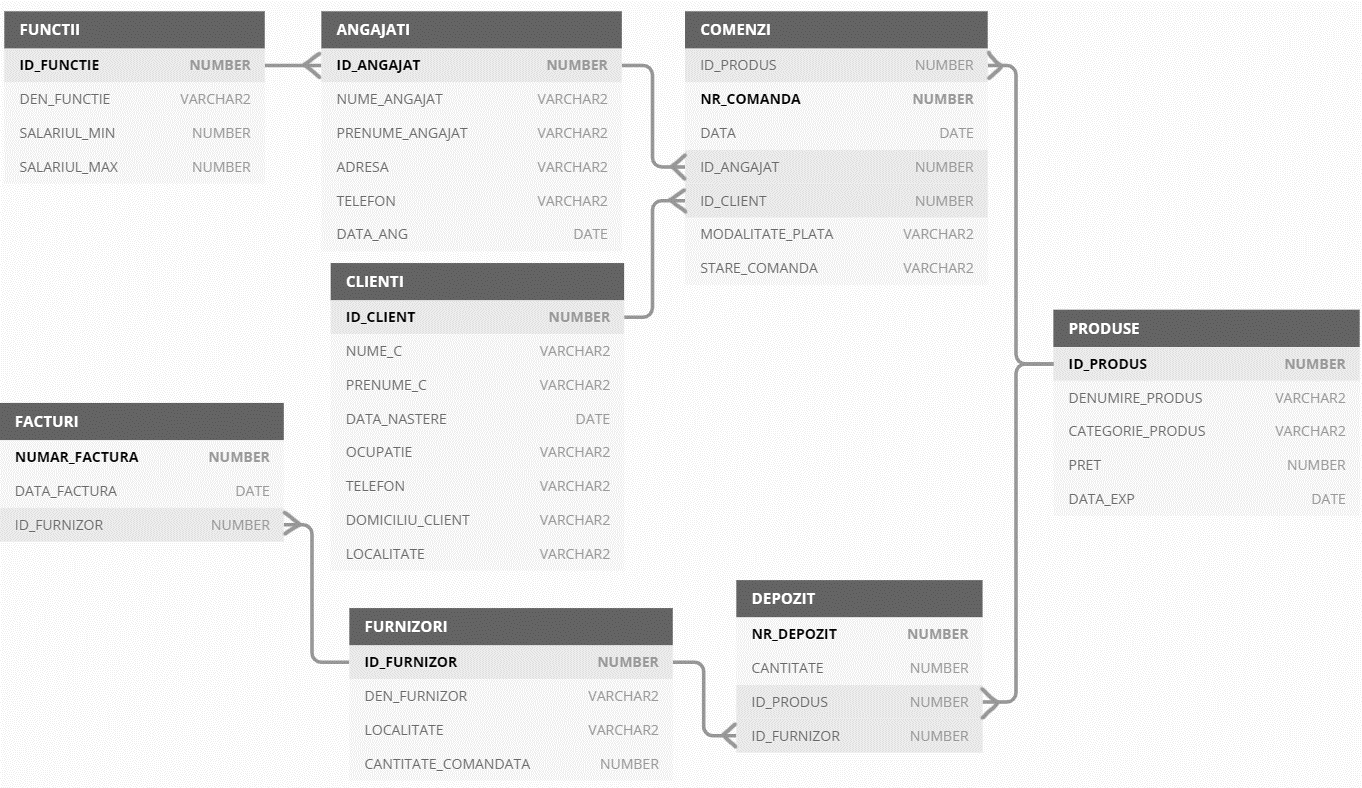
grupa 1055, seria D

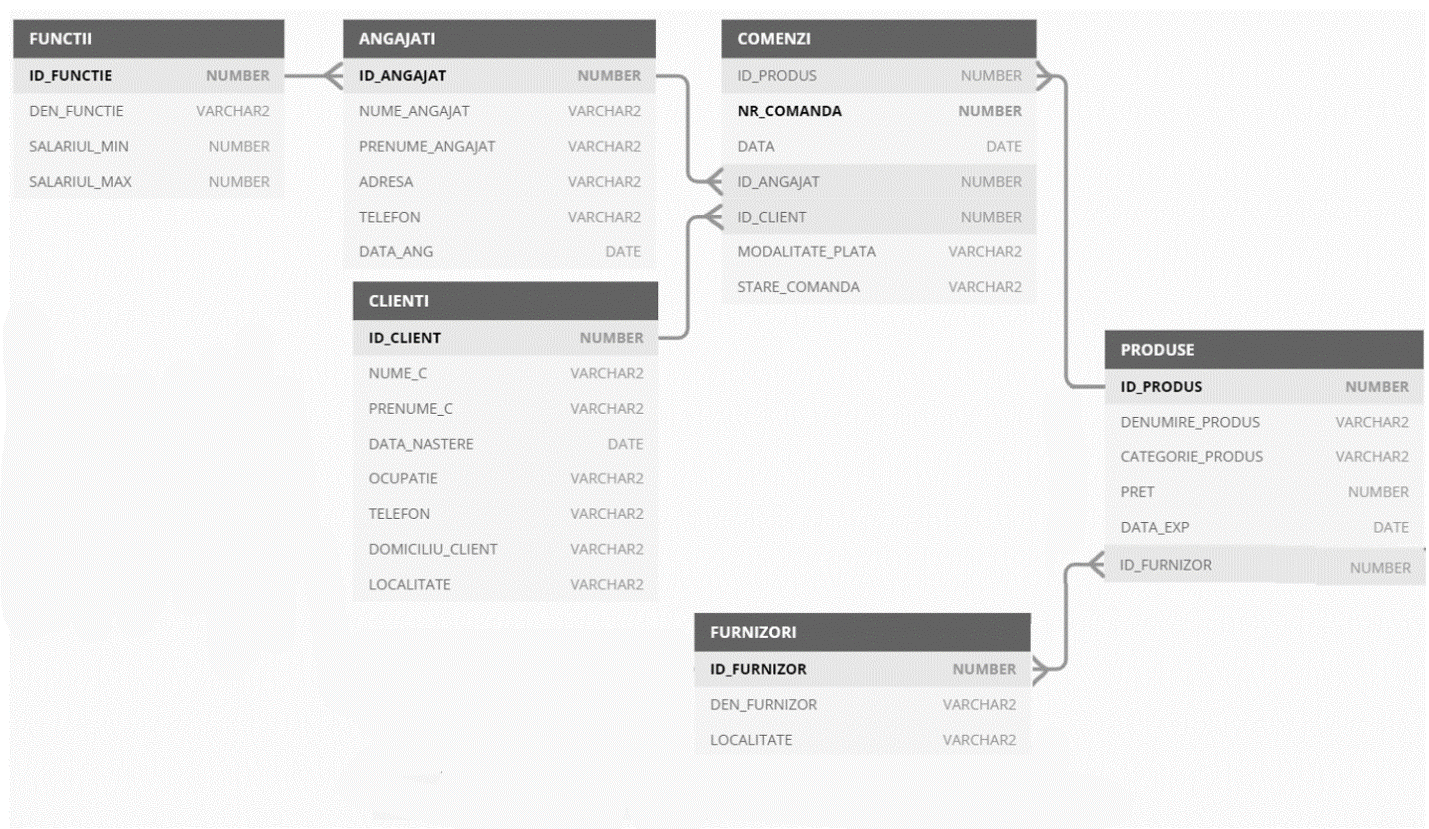
## Descrierea problemei si prezentarea schemei conceptuale a tabelelor

Descrierea temei

A fost creata o baza de date care sa tina evidenta tuturor activitatilor care au loc in magazinul de cosmetice. S-a pus accentul pe subiecte mai importante ale procesului economic, ajungand astfel la urmatoarele tabele: Angajati, Clienti, Comezi, Produse, Functii.

Produsele vandute clientilor prin comenzi preluate de angajati ce au anumite functii. Fiecare produs provine de la cate un furnizor. Un furnizor poate furniza magazinului mai multe produse.

Schema conceptuala din semestrul trecut:

Schema conceptuala din semestrul acesta:

Din schema actuala au fost scoase tabelele FACTURI si DEPOZIT deoarece mi s-au parut a fi irelevante. Astfel noua legatura va fi intre FURNIZORI si PRODUSE. De asemenea am scos din tabela FURNIZORI coloana de Cantitate\_comandata.

## Interactiunea cu serverul Oracle prin intermediul comenzilor SQL

--SA SE STEARGA DIN TABELA CLIENTI COLOANA DATA\_NASTERE

DECLARE QUERY VARCHAR2(150);

BEGIN

    QUERY:='ALTER TABLE CLIENTI DROP COLUMN DATA\_NASTERE';

    EXECUTE IMMEDIATE QUERY;

END;

/

--SA SE MODIFICE PRETUL UNUI PRODUS CU ID INTRODUS DE LA TASTATURA

DECLARE

    VID PRODUSE.ID\_PRODUS%TYPE;

    VPRET PRODUSE.PRET%TYPE;

BEGIN

    VID:=&X;

    VPRET:=&Y;

    UPDATE PRODUSE SET PRET=VPRET WHERE ID\_PRODUS=VID;

END;

/

--SA SE CREEZE TABELA PRODUSE\_5 CE CONTINE DOAR PRODUSELE CU PRETUL = 5

BEGIN

    EXECUTE IMMEDIATE 'CREATE TABLE PRODUSE\_MACHIAJ AS SELECT \* FROM PRODUSE WHERE PRET=5';

END;

/

--SA SE STEARGA COMANDA SAU COMENZILE DIN DATA DE 02.12.2013

BEGIN

    DELETE FROM COMENZI WHERE DATA=TO\_DATE('02.12.2013','DD.MM.YYYY');

END;

/

--SA SE STEARGA PRODUSELE CU PRET MAXIM

DECLARE PRET\_M NUMBER;

BEGIN

    SELECT MAX(PRET) INTO PRET\_M FROM PRODUSE;

    DELETE FROM PRODUSE WHERE PRET=PRET\_M;

    ROLLBACK;

END;

/

## Structuri alternative si repetitive.

--SA SE AFISEZE ANGAJATII CU ID IN INTERVALUL 1 - 10 SI SALARIUL MAJORAT CU 5% PENTRU ANGAJATII CU SALARIUL SUB 2000,

--MAJORAT CU 7% PENTRU ANGAJATII CU SALARIUL INTRE 2000 SI 3000, MAJORAT CU 10% PENTRU ANGAJATII CU SALARIUL PESTE 3000

DECLARE ID NUMBER; SAL NUMBER;

BEGIN

    FOR I IN 1..10 LOOP

        SELECT ID\_ANGAJAT,SALARIUL INTO ID,SAL FROM ANGAJATI WHERE ID\_ANGAJAT=I;

        CASE

            WHEN SAL<2000 THEN

                DBMS\_OUTPUT.PUT\_LINE(ID||' -> '||SAL\*1.05);

            WHEN SAL>=2000 AND SAL<=3000 THEN

                DBMS\_OUTPUT.PUT\_LINE(ID||' -> '||SAL\*1.07);

            WHEN SAL>3000 THEN

                DBMS\_OUTPUT.PUT\_LINE(ID||' -> '||SAL\*1.10);

        END CASE;

    END LOOP;

END;

/

--SA SE AFISEZE PRODUSELE CU PRETUL MAI MARE DECAT CEL MEDIU SI CU ID INTRE 1 SI 5

DECLARE PRET\_M NUMBER; PRET\_P NUMBER; DENUMIRE VARCHAR2(100);ID NUMBER:=1;

BEGIN

    SELECT AVG(PRET) INTO PRET\_M FROM PRODUSE;

    LOOP

        EXIT WHEN ID>5;

        SELECT DENUMIRE\_PRODUS,PRET INTO DENUMIRE,PRET\_P FROM PRODUSE WHERE ID\_PRODUS=ID;

        IF PRET\_P>PRET\_M THEN

            DBMS\_OUTPUT.PUT\_LINE(ID||' -> '||PRET\_P);

        END IF;

        ID:=ID+1;

    END LOOP;

END;

/

--SA SE AFISEZE DACA O COMANDA CU ID INTRODUS DE LA TASTATURA A FOST LIVRATA SAU NU

DECLARE ID NUMBER;STATUS VARCHAR2(20);

BEGIN

    ID:=&X;

    SELECT STARE\_COMANDA INTO STATUS FROM COMENZI WHERE NR\_COMANDA=ID;

    IF UPPER(STATUS)='LIVRAT' THEN

        DBMS\_OUTPUT.PUT\_LINE('COMANDA '||ID||' A FOST LIVRATA!');

    ELSE

        DBMS\_OUTPUT.PUT\_LINE('COMANDA '||ID||' NU A FOST LIVRATA!');

    END IF;

END;

/

## Tratarea exceptiilor

--SA SE AFISEZE NUMARUL DE TELEFON AL UNUI CLIENT CU ID INTRODUS DE LA TASTATURA. SE VA TRATA EXCEPTIA DACA NU EXISTA CLIENTUL CU ID SPECIFICAT

DECLARE ID NUMBER;TELEFON VARCHAR2(13);

BEGIN

    ID:=&X;

    SELECT TELEFON INTO TELEFON FROM CLIENTI WHERE ID\_CLIENT=ID;

    DBMS\_OUTPUT.PUT\_LINE(TELEFON);

    EXCEPTION

        WHEN NO\_DATA\_FOUND THEN

            DBMS\_OUTPUT.PUT\_LINE('ANGAJATUL NU EXISTA!');

END;

/

--SA SE AFISEZE UN PRODUS DIN CATEGORIA UNGHII. SE VA TRATA EXCEPTIA DACA EXISTA MAI MULTE PRODUSE DE ACEST GEN SAU NU EXISTA NICIUNUL

DECLARE DENUMIRE VARCHAR2(100);

BEGIN

    SELECT DENUMIRE\_PRODUS INTO DENUMIRE FROM PRODUSE WHERE UPPER(CATEGORIE\_PRODUS)='UNGHII';

    DBMS\_OUTPUT.PUT\_LINE(DENUMIRE);

    EXCEPTION

        WHEN TOO\_MANY\_ROWS THEN

            DBMS\_OUTPUT.PUT\_LINE('EXISTA MAI MULTE PRODUSE IN ACEASTA CATEGORIE!');

        WHEN NO\_DATA\_FOUND THEN

            DBMS\_OUTPUT.PUT\_LINE('NU EXISTA PRODUSE IN ACEASTA CATEGORIE!');

END;

/

--SA SE INTORDUCA UN FURNIZOR CU ID 100 SI RESTUL ATRIBUTELOR NULL. SE VA TRATA EXCEPTIA DACA NU POT FI INTRODUSE VALORI NULL IN CELELALTE COLOANE

DECLARE NULL\_VALUE EXCEPTION; PRAGMA EXCEPTION\_INIT(NULL\_VALUE,-01400);

BEGIN

    INSERT INTO FURNIZORI(ID\_FURNIZOR) VALUES(100);

    EXCEPTION

        WHEN NULL\_VALUE THEN

            DBMS\_OUTPUT.PUT\_LINE('NU POT FI INTRODUSE VALORI NULL!');

END;

/

--SA SE MAJOREZE CU 5% SALARIILE SUB 1000. SE VA TRATA EXCEPTIA DACA NU EXISTA SALARII SUB 1000

DECLARE FARA\_MODIFICARI EXCEPTION;

BEGIN

    UPDATE ANGAJATI SET SALARIUL=SALARIUL\*1.05 WHERE SALARIUL<1000;

    IF SQL%NOTFOUND THEN

        RAISE FARA\_MODIFICARI;

    END IF;

    EXCEPTION

        WHEN FARA\_MODIFICARI THEN

            DBMS\_OUTPUT.PUT\_LINE('NU EXISTA ANGAJATI CU SALARIUL SUB 1000!');

END;

/

--SA SE STEARGA PRODUSUL CU ID CITIT DE LA TASTATURA

DECLARE NOT\_FOUND EXCEPTION;ID NUMBER;

BEGIN

    ID:=&X;

    DELETE FROM PRODUSE WHERE ID\_PRODUS=ID;

    IF SQL%NOTFOUND THEN

        RAISE NOT\_FOUND;

    END IF;

    EXCEPTION

        WHEN NOT\_FOUND THEN

            DBMS\_OUTPUT.PUT\_LINE('NU EXISTA PRODUSUL CU ID SPECIFICAT!');

END;

## Gestionarea cursorilor

--SA SE AFISEZE PRODUSELE FIECARUI FURNIZOR

DECLARE

    CURSOR CF IS SELECT \* FROM FURNIZORI;

    CURSOR CP(ID NUMBER) IS SELECT \* FROM PRODUSE WHERE ID\_FURNIZOR=ID;

BEGIN

    FOR RF IN CF LOOP

        DBMS\_OUTPUT.PUT\_LINE(RF.DEN\_FURNIZOR||':');

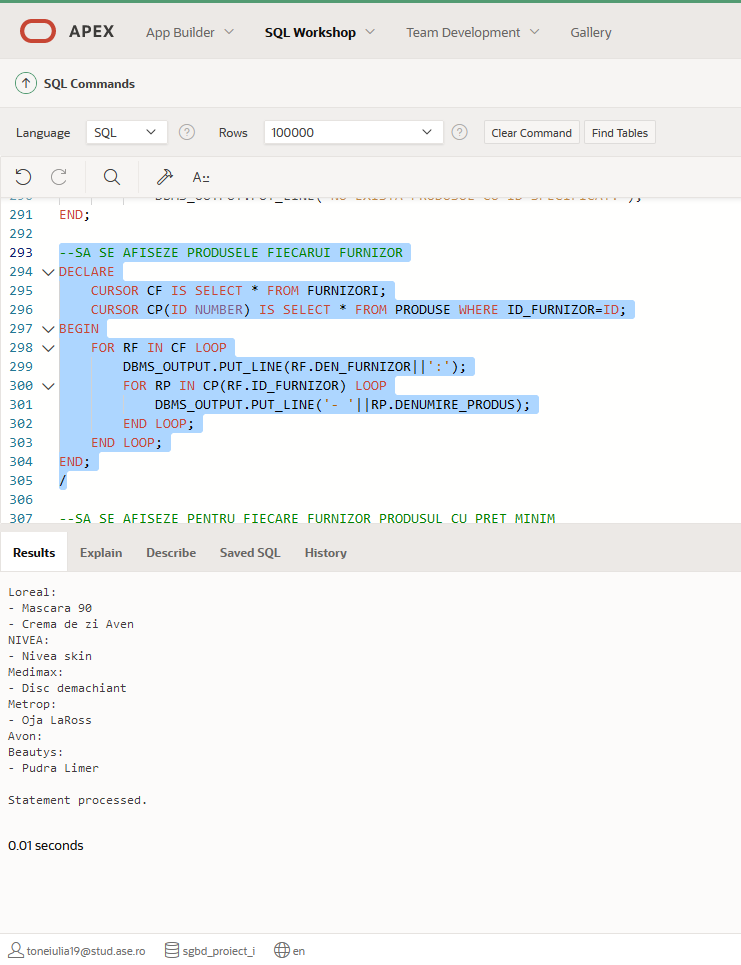
        FOR RP IN CP(RF.ID\_FURNIZOR) LOOP

            DBMS\_OUTPUT.PUT\_LINE('- '||RP.DENUMIRE\_PRODUS);

        END LOOP;

    END LOOP;

END;



--SA SE AFISEZE PENTRU FIECARE FURNIZOR PRODUSUL CU PRET MINIM

DECLARE

    CURSOR CF IS SELECT \* FROM FURNIZORI;

    CURSOR CP(PRET NUMBER, ID NUMBER) IS SELECT \* FROM PRODUSE WHERE PRET=PRET AND ID\_FURNIZOR=ID;

    PRET NUMBER;

BEGIN

    FOR RF IN CF LOOP

        DBMS\_OUTPUT.PUT\_LINE(RF.DEN\_FURNIZOR||':');

        SELECT MIN(PRET) INTO PRET FROM PRODUSE WHERE ID\_FURNIZOR=RF.ID\_FURNIZOR;

        FOR RP IN CP(PRET, RF.ID\_FURNIZOR) LOOP

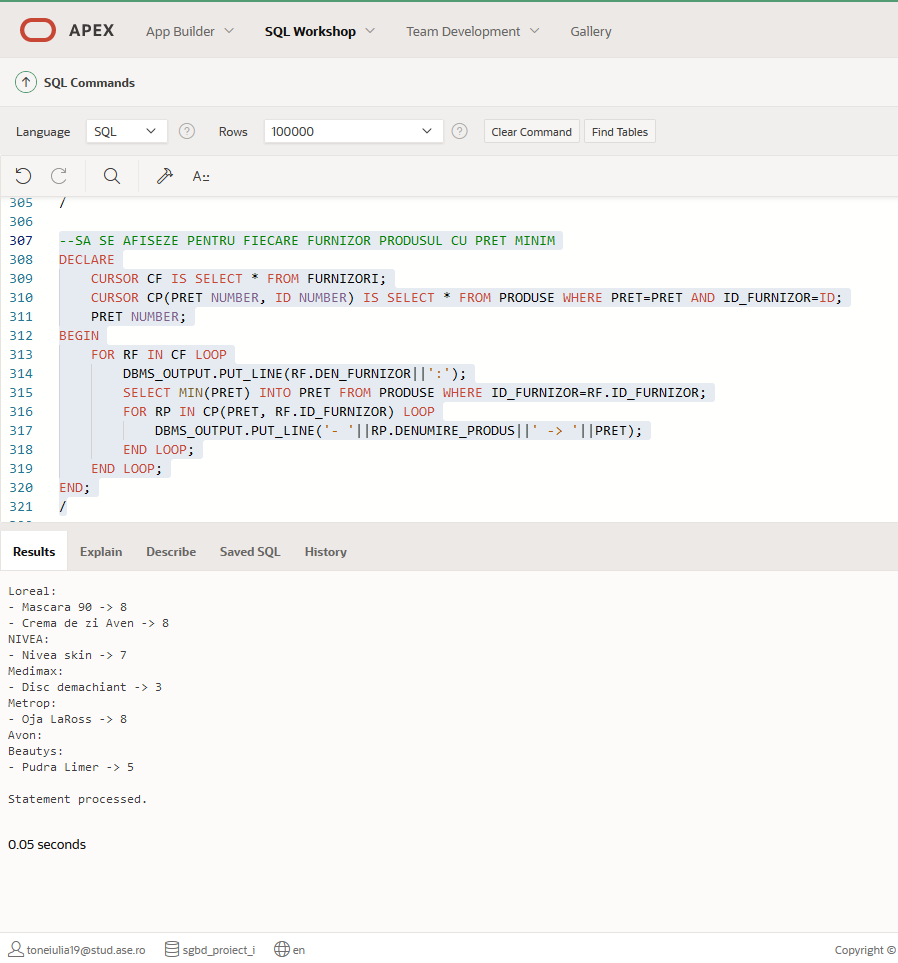
            DBMS\_OUTPUT.PUT\_LINE('- '||RP.DENUMIRE\_PRODUS||' -> '||PRET);

        END LOOP;

    END LOOP;

END;

/



--SA SE MICSOREZE CU 5% PRETUL PRODUSELOR DIN CATEGORIA NECESARE SI SA SE AFISEZE NUMARUL DE MODIFICARI EFECTUATE

BEGIN

    UPDATE PRODUSE SET PRET=PRET\*0.95 WHERE UPPER(CATEGORIE\_PRODUS)='NECESARE';

    IF SQL%NOTFOUND THEN

        DBMS\_OUTPUT.PUT\_LINE('NU A FOST EFECTUATA NICIO MODIFICARE');

    ELSE

        DBMS\_OUTPUT.PUT\_LINE('AU FOST EFECTUATE '||SQL%ROWCOUNT||' MODIFICARI');

    END IF;

END;

/

--SA SE STEARGA COMENZILE ANGAJATULUI CU ID 3 SI SA SE AFISEZE NUMARUL DE STERGERI EFECTUATE

BEGIN

    DELETE FROM COMENZI WHERE ID\_ANGAJAT=3;

    IF SQL%NOTFOUND THEN

        DBMS\_OUTPUT.PUT\_LINE('NU A FOST EFECTUATA NICIO STERGERE');

    ELSE

        DBMS\_OUTPUT.PUT\_LINE('AU FOST EFECTUATE '||SQL%ROWCOUNT||' STERGERI');

    END IF;

END;

/

--SA SE AFISEZE PRIMII 3 CLIENTI IN ORDINE DESCRESCATOARE A NUMARULUI DE COMENZI

DECLARE

    CURSOR C IS SELECT ID\_CLIENT, COUNT(ID\_CLIENT) NR\_COMENZI FROM COMENZI GROUP BY ID\_CLIENT ORDER BY 2 DESC;

    R C%ROWTYPE;

BEGIN

    OPEN C;

    LOOP

        FETCH C INTO R;

        EXIT WHEN C%NOTFOUND OR C%ROWCOUNT>3;

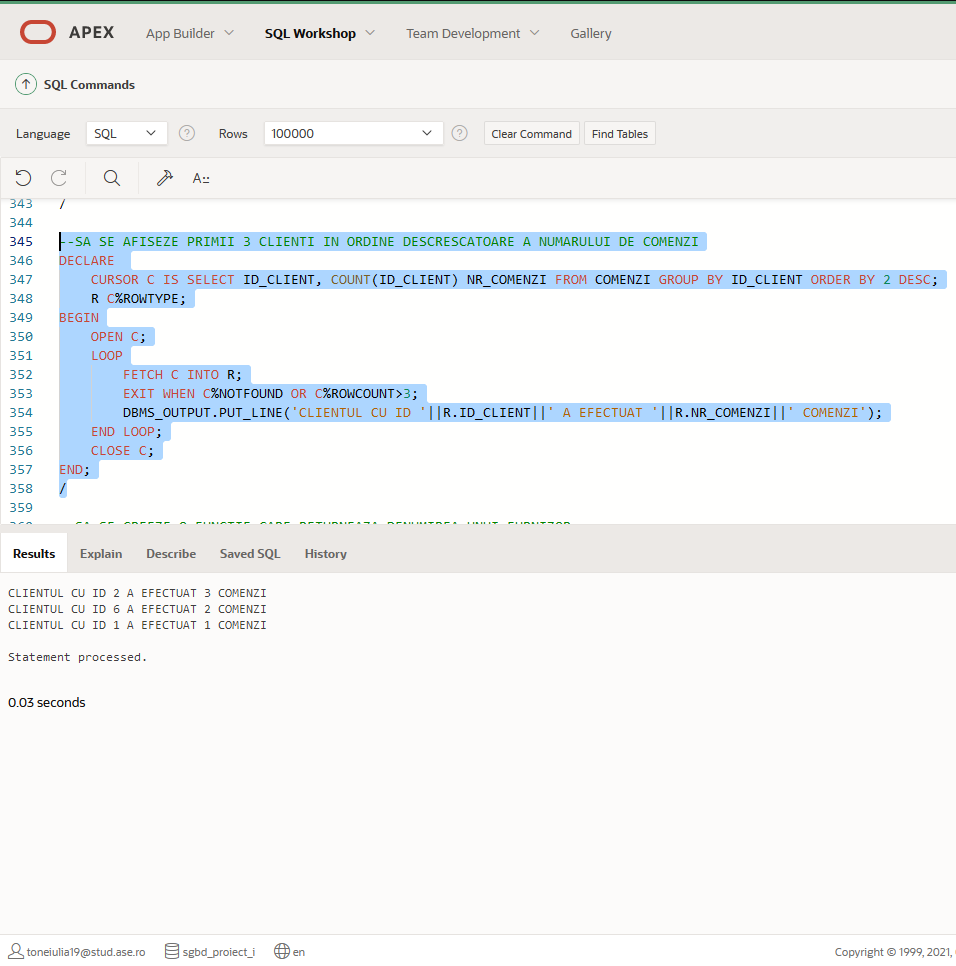
        DBMS\_OUTPUT.PUT\_LINE('CLIENTUL CU ID '||R.ID\_CLIENT||' A EFECTUAT '||R.NR\_COMENZI||' COMENZI');

    END LOOP;

    CLOSE C;

END;

/



## Functii, proceduri, includerea acestora in pachete

--SA SE CREEZE O FUNCTIE CARE RETURNEAZA DENUMIREA UNUI FURNIZOR

CREATE OR REPLACE FUNCTION DEN\_F(ID IN NUMBER, DEN OUT VARCHAR2) RETURN VARCHAR2 IS

BEGIN

    SELECT DEN\_FURNIZOR INTO DEN FROM FURNIZORI WHERE ID\_FURNIZOR=ID;

    RETURN DEN;

END;

--SA SE CREEZE O FUNCTIE CARE RETURNEAZA SALARIUL MAXIM

CREATE OR REPLACE FUNCTION SAL\_MAX(SAL OUT NUMBER) RETURN NUMBER IS

BEGIN

    SELECT MAX(SALARIUL) INTO SAL FROM ANGAJATI;

    RETURN SAL;

END;

/

--SA SE CREEZE O FUNCTIE CE VERIFICA DACA EXISTA UN ANGAJAT CU UN SALARIU MAI MARE DECAT CEL FURNIZAT CA PARAMETRU

CREATE OR REPLACE FUNCTION SAL(VAL IN NUMBER) RETURN BOOLEAN IS

SAL\_MAX NUMBER;

BEGIN

    SELECT MAX(SALARIUL) INTO SAL\_MAX FROM ANGAJATI;

    IF SAL\_MAX > VAL THEN

        RETURN TRUE;

    ELSE

        RETURN FALSE;

    END IF;

    EXCEPTION

        WHEN NO\_DATA\_FOUND THEN

            RETURN NULL;

END;

/

--SA SE CREEZE O PROCEDURA CE STERGE UN PRODUS CU ID FURNIZAT

CREATE OR REPLACE PROCEDURE DEL\_PROD(ID IN NUMBER) IS

BEGIN

    DELETE FROM PRODUSE WHERE ID\_PRODUS=ID;

END;

/

--SA SE CREEZE O PROCEDURA CE AFISEAZA CLIENTII CARE NU AU EFECTUAT COMENZI

CREATE OR REPLACE PROCEDURE COMENZI\_CL IS

CURSOR C IS SELECT NUME\_C||' '||PRENUME\_C NUME\_PRENUME FROM CLIENTI WHERE ID\_CLIENT NOT IN(SELECT ID\_CLIENT FROM PRODUSE);

BEGIN

    FOR R IN C LOOP

        DBMS\_OUTPUT.PUT\_LINE(R.NUME\_PRENUME);

    END LOOP;

END;

/

--SA SE CREEZE UN PACHET CE CONTINE PROCEDURA COMENZI\_CL CREATA ANTERIOR

CREATE OR REPLACE PACKAGE PACHET IS

    PROCEDURE COMENZI\_CL;

END;

/

CREATE OR REPLACE PACKAGE BODY PACHET IS

    PROCEDURE COMENZI\_CL IS

    CURSOR C IS SELECT NUME\_C||' '||PRENUME\_C NUME\_PRENUME FROM CLIENTI WHERE ID\_CLIENT NOT IN(SELECT ID\_CLIENT FROM PRODUSE);

    BEGIN

        FOR R IN C LOOP

            DBMS\_OUTPUT.PUT\_LINE(R.NUME\_PRENUME);

        END LOOP;

    END;

END;

/

## Declansatori

--SA SE CREEZE UN TRIGGER CARE SA SE DECLANSEZE DACA SE INTRODUC PRODUSE CU PRETUL PESTE 90000

CREATE OR REPLACE TRIGGER PRET\_TR

    BEFORE INSERT OR UPDATE ON PRODUSE

    FOR EACH ROW

        DECLARE

            PRET\_MAX NUMBER :=90000;

        BEGIN

            IF:NEW.PRET<PRET\_MAX THEN

                RAISE\_APPLICATION\_ERROR(-20000,'PRET INVALID!');

            END IF;

        END;

--SA SE CREEZE UN TRIGGER CARE SE DECLANSEAZA LA MODIFICAREA MODALITATII DE PLATA PENTRU O COMANDA

CREATE OR REPLACE TRIGGER MODC\_TR

    BEFORE UPDATE OF MODALITATE\_PLATA ON COMENZI

    BEGIN

        IF UPDATING ('MODALITATE\_PLATA') THEN

            RAISE\_APPLICATION\_ERROR(-20000,'MODALITATEA DE PLATA NU POATE FI MODIFICATA!');

        END IF;

    END;

--SA SE CREEZE UN TRIGGER CARE SE DECLANSEAZA LA MODIFICAREA NUMELUI SAU PRENUMELUI ANGAJATULUI CU ID 1

CREATE OR REPLACE TRIGGER ANG1\_TR

    BEFORE UPDATE ON ANGAJATI

    FOR EACH ROW

        DECLARE ID NUMBER;

        BEGIN

            SELECT ID\_ANGAJAT INTO ID FROM ANGAJATI;

            IF (UPDATING ('NUME') OR UPDATING ('NUME')) AND ID=1 THEN

                RAISE\_APPLICATION\_ERROR(-20000,'NUMELE SI PRENUMELUI ANGAJATULUI CU ID 1 NU POATE FI MODIFICAT!');

            END IF;

        END;

--SA SE CREEZE UN TRIGGER CARE NU PERMITE INSERAREA DE PRODUSE IN INTERVALUL ORAR 00:00 - 01:00

CREATE OR REPLACE TRIGGER PROD\_H

    BEFORE INSERT ON PRODUSE

    BEGIN

        IF INSERTING AND (TO\_CHAR(SYSDATE, 'HH24:MI') > '00:00' AND TO\_CHAR(SYSDATE, 'HH24:MI') < '01:00') THEN

            RAISE\_APPLICATION\_ERROR(-20000, 'NU POT FI INSERATE PRODUSE IN ACEST INTERVAL ORAR!');

        END IF;

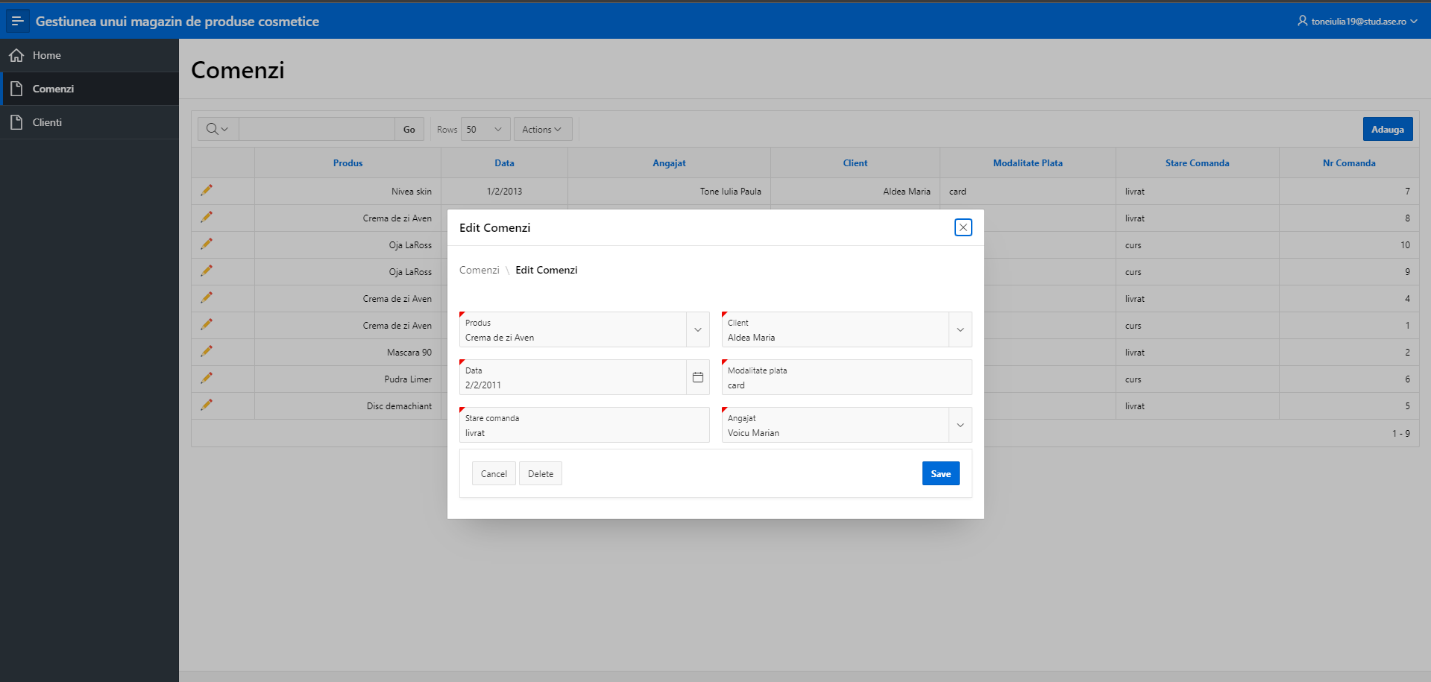
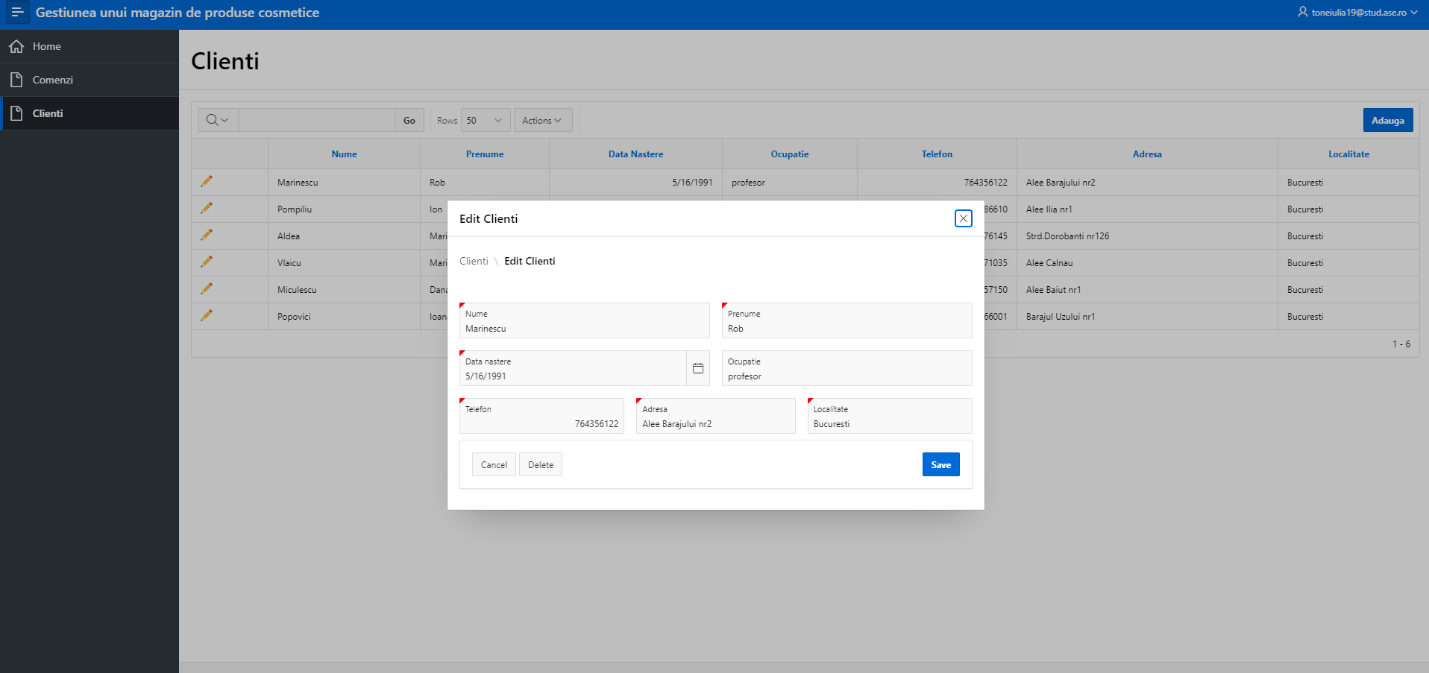
    END;

## Machete de intrare apex

Link aplicatie APEX: <https://apex.oracle.com/pls/apex/sgbd_proiect_i/r/gestiunea-unui-magazin-de-produse-cosmetice/home>

Username: [toneiulia19@stud.ase.ro](mailto:toneiulia19@stud.ase.ro)

Password: elpatron1



## Machete de iesire apex

