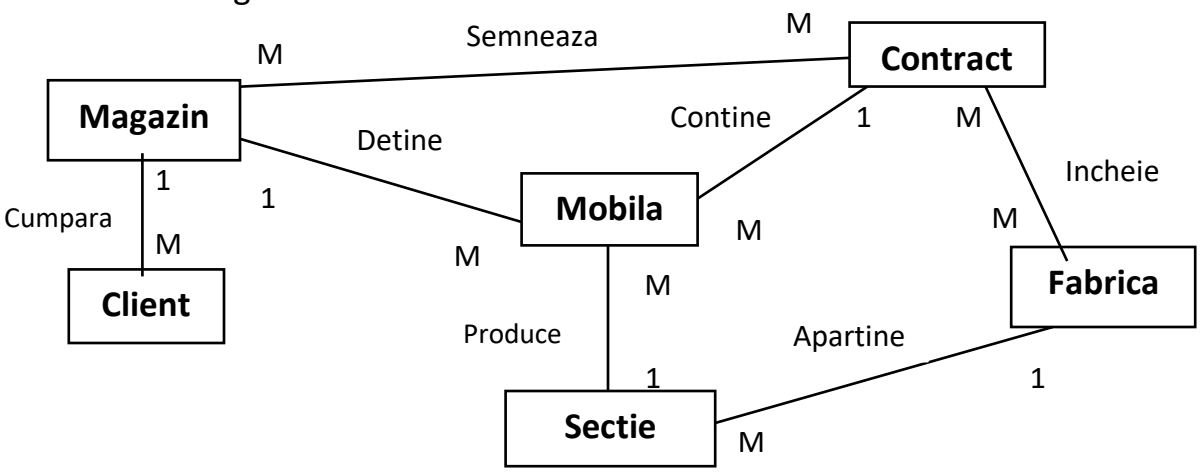
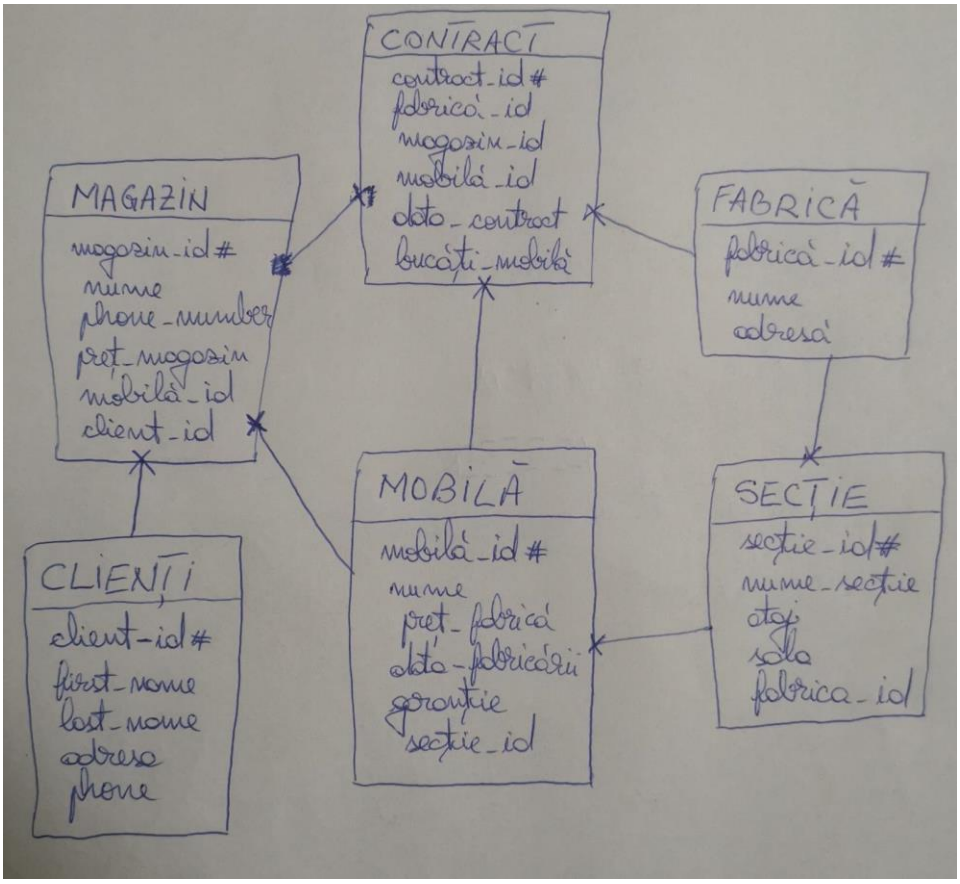
Proiect Arsene Marinel – grupa 233

1. Baza mea de date contine 6 tabele: client, magazin, mobila, sectie, fabrica si contract. Dintre acestea 5 tabele sunt independente: client, magazin, mobila, sectie, fabrica; iar o tabela este asociativa: cotract. Tabele au urmatorul scop:
    - Client – contine clientii magazinului
    - Magazin – contine mobila provenita din fabrica
    - Sectie – contine mobila produsa in sectie
    - Fabrica – contine mai multe sectii
    - Contract – contine toate contractele incheiate intre fabrica si magazin

2. Diagrama entitate-relatie:



3. Diagrama conceptuala:



4. Implementarea diagramei conceptuale:

```sql
create table CLIENTI(
    client_id number PRIMARY KEY,
    first_name VARCHAR2(20) not null,
    last_name VARCHAR2(20) not null,
    adresa VARCHAR2(30) not null,
    phone VARCHAR2(10) not null
);

create table FABRICA(
    fabrica_id number PRIMARY KEY,
    nume VARCHAR(20) not null,
    adresa VARCHAR(30) not null
);

create table SECTIE(
    sectie_id number PRIMARY KEY,
    nume_sectie VARCHAR(20) not null,
    etaj number not null,
    sala number(3) not null,
    fabrica_id number not null
);

create table MOBILA(
    mobila_id number PRIMARY KEY,
    nume VARCHAR2(20) not null,
    pret_fabrica number(20) not null,
    data_fabricarii date not null,
    garantie number(2) not null,
    sectie_id number not null
);
```

```sql
create table MAGAZIN(
    magazin_id number PRIMARY KEY,
    nume NVARCHAR2(20) not null,
    phone_number VARCHAR2(10),
    pret_magazin NUMBER(20) not null,
    mobila_id number not null,
    client_id number not null
);

create table CONTRACT(
    contract_id number PRIMARY KEY,
    fabrica_id number not null,
    magazin_id number not null,
    mobila_id number not null,
    data_contract date not null,
    bucati_mobila number(10) not null,
    constraint fk_fabrica
        foreign key (fabrica_id)
        references fabrica(fabrica_id),
    constraint fk_magazin
        foreign key (magazin_id)
        references magazin(magazin_id),
     constraint fk_mobila
        foreign key (mobila_id)
        references mobila(mobila_id)
);
```

Script Output

```
Table CLIENTI created.

Table FABRICA created.

Table SECTIE created.

Table MOBILA created.

Table MAGAZIN created.

Table CONTRACT created.
```

5. Adaug clientii:

```sql
INSERT INTO CLIENTI (client_id, first_name, last_name, adresa, phone)
VALUES (1, 'ISTRATE', 'ROBERT', 'MIRCEA', '0123456789');
INSERT INTO CLIENTI (client_id, first_name, last_name, adresa, phone)
VALUES (2, 'POPESCU', 'GEORGEL', 'VODA', '0123456798');
INSERT INTO CLIENTI (client_id, first_name, last_name, adresa, phone)
VALUES (3, 'POPESCU', 'FLORENTIN', 'STEFAN_CEL_MARE', '0123456978');
INSERT INTO CLIENTI (client_id, first_name, last_name, adresa, phone)
VALUES (4, 'GEORGESCU', 'MATEI', 'EMINESCU', '0123459678');
INSERT INTO CLIENTI (client_id, first_name, last_name, adresa, phone)
VALUES (5, 'ALBERT', 'FILIP', 'CARAGIALE', '9876543210');
```

Script Output | Query Result

SQL | All Rows Fetched: 4 in 0.122 seconds

	CLIENT_ID	FIRST_NAME	LAST_NAME	ADRESA	PHONE
1	1	ISTRATE	ROBERT	MIRCEA	0123456789
2	2	POPESCU	GEORGEL	VODA	0123456798
3	4	GEORGESCU	MATEI	EMINESCU	0123459678
4	5	ALBERT	FILIP	CARAGIALE	9876543210

Adaug magazinele:

```
--Magazin:
INSERT INTO MAGAZIN(magazin_id, nume, phone_number, pret_magazin, client_id, mobila_id)
VALUES (1, 'PRIMA_MANA', '1230984378', 12345, 2, 1);
INSERT INTO MAGAZIN(magazin_id, nume, phone_number, pret_magazin, client_id, mobila_id)
VALUES (2, 'CALITATE', '9999999999', 65151, 1, 2);
INSERT INTO MAGAZIN(magazin_id, nume, phone_number, pret_magazin, client_id, mobila_id)
VALUES (3, 'JOHN', '8888888888', 4565, 3, 1);
INSERT INTO MAGAZIN(magazin_id, nume, phone_number, pret_magazin, client_id, mobila_id)
VALUES (4, 'PRIETENUL_TAU', '1111111111', 462, 5, 4);
INSERT INTO MAGAZIN(magazin_id, nume, phone_number, pret_magazin, client_id, mobila_id)
VALUES (5, 'LANGA_TINE', '2222222222', 564216, 4, 1);

select *
from magazin;

--Mobila:
```

Script Output × | Query Result ×

SQL | All Rows Fetched: 5 in 0.068 seconds

	MAGAZIN_ID	NUME	PHONE_NUMBER	PRET_MAGAZIN	MOBILA_ID	CLIENT_ID
1	1	PRIMA_MANA	1230984378	12345	1	2
2	2	CALITATE	9999999999	65151	2	1
3	3	JOHN	8888888888	4565	1	3
4	4	PRIETENUL_TAU	1111111111	462	4	5
5	5	LANGA_TINE	2222222222	564216	1	4

Adaug mobila:

```
INSERT INTO MOBILA(mobila_id, nume, pret_fabrica, data_fabricarii, garantie, sectie_id)
VALUES (1, 'DULAP', 1800, TO_DATE('20-MAR-2010', 'dd-MON-yyyy'), 2, 1);
INSERT INTO MOBILA(mobila_id, nume, pret_fabrica, data_fabricarii, garantie, sectie_id)
VALUES (2, 'NOPTIERA', 200, TO_DATE('20-MAR-2012', 'dd-MON-yyyy'), 2, 1);
INSERT INTO MOBILA(mobila_id, nume, pret_fabrica, data_fabricarii, garantie, sectie_id)
VALUES (3, 'COMODA', 1000, TO_DATE('10-MAR-2010', 'dd-MON-yyyy'), 2, 1);
INSERT INTO MOBILA(mobila_id, nume, pret_fabrica, data_fabricarii, garantie, sectie_id)
VALUES (4, 'SIFONIER', 2200, TO_DATE('15-MAY-2015', 'dd-MON-yyyy'), 2, 1);
INSERT INTO MOBILA(mobila_id, nume, pret_fabrica, data_fabricarii, garantie, sectie_id)
VALUES (5, 'DULAP', 222, TO_DATE('20-MAR-2010', 'dd-MON-yyyy'), 2, 1);
```

Script Output × | Query Result ×

SQL | All Rows Fetched: 5 in 0.083 seconds

	MOBILA_ID	NUME	PRET_FABRICA	DATA_FABRICARII	GARANTIE	SECTIE_ID
1	1	DULAP	1800	20-MAR-10	2	1
2	2	NOPTIERA	200	20-MAR-12	2	1
3	3	COMODA	1000	10-MAR-10	2	1
4	4	SIFONIER	2200	15-MAY-15	2	1
5	5	DULAP	222	20-MAR-10	2	1

Adaug sectie:

```
INSERT INTO SECTIE(sectie_id, nume_sectie, etaj, sala, fabrica_id)
VALUES (1, 'MOBILA', 2, 205, 1);
INSERT INTO SECTIE(sectie_id, nume_sectie, etaj, sala, fabrica_id)
VALUES (2, 'VASE', 1, 101, 1);
INSERT INTO SECTIE(sectie_id, nume_sectie, etaj, sala, fabrica_id)
VALUES (3, 'METALURGIE', 3, 306, 1);
INSERT INTO SECTIE(sectie_id, nume_sectie, etaj, sala, fabrica_id)
VALUES (4, 'LUSTRE', 4, 404, 1);
INSERT INTO SECTIE(sectie_id, nume_sectie, etaj, sala, fabrica_id)
VALUES (5, 'GRADINARIT', 5, 501, 1);
```

Script Output × Query Result ×

SQL | All Rows Fetched: 5 in 0.06 seconds

	SECTIE_ID	NUME_SECTIE	ETAJ	SALA	FABRICA_ID
1	1	MOBILA	2	205	1
2	2	VASE	1	101	1
3	3	METALURGIE	3	306	1
4	4	LUSTRE	4	404	1
5	5	GRADINARIT	5	501	1

Adaug fabrica:

```
INSERT INTO FABRICA(fabrica_id, nume, adresa)
VALUES (1, 'LEONI', 'CARPATI');
INSERT INTO FABRICA(fabrica_id, nume, adresa)
VALUES (2, 'SBP', 'BARBARII');
INSERT INTO FABRICA(fabrica_id, nume, adresa)
VALUES (3, 'MIX', 'NOBILII');
INSERT INTO FABRICA(fabrica_id, nume, adresa)
VALUES (4, 'S.U.P.E.R', 'EVOLUTIEI');
INSERT INTO FABRICA(fabrica_id, nume, adresa)
VALUES (5, 'CALITATE', 'REVOLUTIEI');
```

Script Output × Query Result ×

SQL | All Rows Fetched: 5 in 0.073 seconds

	FABRICA_ID	NUME	ADRESA
1	1	LEONI	CARPATI
2	2	SBP	BARBARII
3	3	MIX	NOBILII
4	4	S.U.P.E.R	EVOLUTIEI
5	5	CALITATE	REVOLUTIEI

Adaug contract:

```
INSERT INTO CONTRACT(contract_id, fabrica_id, magazin_id, mobila_id, data_contract, bucati_mobila)
VALUES (1, 1, 1, 1, '23-SEP-2018', 200);
INSERT INTO CONTRACT(contract_id, fabrica_id, magazin_id, mobila_id, data_contract, bucati_mobila)
VALUES (2, 1, 1, 2, '24-SEP-2018', 500);
INSERT INTO CONTRACT(contract_id, fabrica_id, magazin_id, mobila_id, data_contract, bucati_mobila)
VALUES (3, 1, 1, 3, '25-SEP-2018', 1000);
INSERT INTO CONTRACT(contract_id, fabrica_id, magazin_id, mobila_id, data_contract, bucati_mobila)
VALUES (4, 1, 1, 4, '26-SEP-2018', 600);
INSERT INTO CONTRACT(contract_id, fabrica_id, magazin_id, mobila_id, data_contract, bucati_mobila)
VALUES (5, 1, 1, 5, '27-SEP-2018', 1200);
INSERT INTO CONTRACT(contract_id, fabrica_id, magazin_id, mobila_id, data_contract, bucati_mobila)
VALUES (6, 1, 1, 1, '28-SEP-2018', 400);
INSERT INTO CONTRACT(contract_id, fabrica_id, magazin_id, mobila_id, data_contract, bucati_mobila)
VALUES (7, 1, 1, 2, '29-SEP-2018', 700);
INSERT INTO CONTRACT(contract_id, fabrica_id, magazin_id, mobila_id, data_contract, bucati_mobila)
VALUES (8, 1, 1, 3, '30-SEP-2018', 1700);
INSERT INTO CONTRACT(contract_id, fabrica_id, magazin_id, mobila_id, data_contract, bucati_mobila)
VALUES (9, 1, 1, 4, '22-SEP-2018', 100);
INSERT INTO CONTRACT(contract_id, fabrica_id, magazin_id, mobila_id, data_contract, bucati_mobila)
VALUES (10, 1, 1, 5, '21-SEP-2018', 300);
```

Script Output | Query Result

SQL | All Rows Fetched: 10 in 0.069 seconds

	CONTRACT_ID	FABRICA_ID	MAGAZIN_ID	MOBILA_ID	DATA_CONTRACT	BUCATI_MOBILA
1	1	1	1	1	23-SEP-18	200
2	2	1	1	2	24-SEP-18	500
3	3	1	1	3	25-SEP-18	1000
4	4	1	1	4	26-SEP-18	600
5	5	1	1	5	27-SEP-18	1200
6	6	1	1	1	28-SEP-18	400
7	7	1	1	2	29-SEP-18	700
8	8	1	1	3	30-SEP-18	1700
9	9	1	1	4	22-SEP-18	100
10	10	1	1	5	21-SEP-18	300

6. Am creat o procedura care primeste ca parametru adresa (care este strada de fapt) unui client si afiseaza numele complet al tuturor clientilor care locuiesc pe acea strada.

```
CREATE OR REPLACE PROCEDURE ex6(adresa_in IN NVARCHAR2)
IS
    TYPE tablou IS TABLE OF CLIENTI%ROWTYPE;
    client_by_adresa tablou;
    total integer;
BEGIN
    DBMS_OUTPUT.enable;
    select *
    bulk collect into client_by_adresa
    from clienti
    where adresa = adresa_in;

    total := client_by_adresa.COUNT;
    FOR i in 1 .. total loop
        dbms_output.put_line(client_by_adresa(i).first_name || ' ' || client_by_adresa(i).last_name);
    end loop;
END;
/

EXECUTE ex6('MIRCEA');
```

```
Procedure EX6 compiled

ISTRATE ROBERT


PL/SQL procedure successfully completed.
```

7. Am creat o procedura care primeste ca parametru un numar ce reprezinta un pret si foloseste un cursor care contine tot mobilierul cu pret_fabrica mai mare decat acest numar pentu a afisa numele si pret_fabrica a mobilierelor.

```
CREATE OR REPLACE PROCEDURE ex7(pret_in in number)
IS
    CURSOR mobila_cursor is
        select *
        from mobila
        where pret_fabrica >= pret_in;
BEGIN
    DBMS_OUTPUT.enable;
    FOR i in mobila_cursor loop
        dbms_output.put_line('Mobilierul: ' || i.nume || ' costa ' || i.pret_fabrica);
    end loop;
END;
/

EXECUTE ex7(600);
```

Script Output × Query Result ×

Task completed in 0.385 seconds

```
Procedure EX7 compiled

Mobilierul: DULAP costa 1800
Mobilierul: COMODA costa 1000
Mobilierul: SIFONIER costa 2200


PL/SQL procedure successfully completed.
```

8. Am creat o functie care primeste ca parametru mobila_id si afiseaza pretul din fabrica al produsului care se gaseste in magazin si contract.

```
CREATE OR REPLACE FUNCTION ex8(mob_id mobila.mobila_id%TYPE)
    return NUMBER
IS
   pret mobila.pret_fabrica%TYPE;
BEGIN
    select pret_magazin
    into pret
    from mobila mo, magazin m, clienti c
    where mo.mobila_id = mob_id and mo.mobila_id = m.mobila_id and c.client_id = m.client_id;
    return pret;

     EXCEPTION
         WHEN no_data_found THEN
             raise_application_error(-20000, 'Nu exista mobila!');
         WHEN too_many_rows THEN
             raise_application_error(-20001, 'Prea multe linii returnate!');
        WHEN others THEN
            raise_application_error(-20002, 'Error!');
END ex8;
/

--Apel fara exceptii
BEGIN
    dbms_output.put_line('Clientul este: ' || ex8(4));
END;
/
```

```
Script Output x  Query Result x
Task completed in 0.353 seconds
PL/SQL procedure successfully completed.


Function EX8 compiled


PL/SQL procedure successfully completed.
```

Apel no_date_found:

```
--Apel no_data_found:
BEGIN
    dbms_output.put_line('Clientul este: ' || ex8(3));
END;
/
```

```
Script Output x  Query Result x
Task completed in 0.375 seconds
ORA-20000: Nu exista mobila!
ORA-06512: at "GRUPA233.EX8", line 14
ORA-06512: at line 2
20000. 00000 -  "%s"
*Cause:    The stored procedure 'raise_application_error'
           was called which causes this error to be generated.
*Action:   Correct the problem as described in the error message or contact
           the application administrator or DBA for more information.
```

Apel too_many_rows:

```
--Apel too_many_rows:
BEGIN
    dbms_output.put_line('Clientul este: ' || ex8(1));
END;
/
```

Script Output | Query Result

Task completed in 0.374 seconds

```
Error starting at line : 229 in command -
BEGIN
    dbms_output.put_line('Clientul este: ' || ex8(1));
END;
Error report -
ORA-20001: Prea multe linii returnate!
ORA-06512: at "GRUPA233.EX8", line 16
ORA-06512: at line 2
```

9. Am creat o procedura care afiseaza pretul_magazin al mobilei care a fost achizitionata de un client, care se gaseste in tabelele mobila, magazin, contract si care a fost produsa in sectia de mobila(id = 1).

```
CREATE OR REPLACE PROCEDURE ex_9(mob_id mobila.mobila_id%TYPE)
IS
   pret magazin.pret_magazin%TYPE;
BEGIN
    select pret_magazin
    into pret
    from mobila mo, magazin m, clienti c, contract co, (select * from sectie where sectie_id = 1)
    where mo.mobila_id = mob_id and mo.mobila_id = m.mobila_id and c.client_id = m.client_id
            and co.mobila_id = m.mobila_id;
    DBMS_OUTPUT.PUT('Pretul este: ' || pret);

     EXCEPTION
         WHEN no_data_found THEN
             raise_application_error(-20000, 'Nu exista mobila!');
         WHEN too_many_rows THEN
             raise_application_error(-20001, 'Prea multe linii returnate!');
        WHEN others THEN
            raise_application_error(-20002, 'Error!');
END;
/
```

Script Output | Query Result

Task completed in 0.551 seconds

```
*Action:

Procedure EX_9 compiled
```

Apel too_many_rows:

```
--Apel too_many_rows:
EXECUTE ex_9(4);
```

Script Output | Query Result | Task completed in 0.462 seconds

```
Error starting at line : 270 in command -
BEGIN ex_9(4); END;
Error report -
ORA-20001: Prea multe linii returnate!
ORA-06512: at "GRUPA233.EX_9", line 16
ORA-06512: at line 1
```

Apel no_data_found:

```
--Apel no_data_found:
EXECUTE ex_9(3);
```

Script Output | Query Result | Task completed in 0.428 seconds

```
ORA-20000: Nu exista mobila!
ORA-06512: at "GRUPA233.EX_9", line 14
ORA-06512: at line 1
20000. 00000 -  "%s"
*Cause:    The stored procedure 'raise_application_error'
           was called which causes this error to be generated.
*Action:   Correct the problem as described in the error message or contact
           the application administrator or DBA for more information.
```

10. Am creat un trigger LMD la nivel de comanda care afiseaza numele user-ului care face modificari asupra tabelului MOBILA.

```sql
CREATE OR REPLACE TRIGGER ex10
    AFTER INSERT OR DELETE OR UPDATE ON MOBILA
DECLARE
    username varchar2(10);
BEGIN
    DBMS_OUTPUT.enable;

    select user
    into username
    from dual;

    if INSERTING then
        DBMS_OUTPUT.PUT_LINE('User-ul ' || username || ' a facut insert in tabela MOBILA');
    elsif DELETING then
        DBMS_OUTPUT.PUT_LINE('User-ul ' || username || ' a sters in tabela MOBILA');
    else
        DBMS_OUTPUT.PUT_LINE('User-ul ' || username || ' a facut update in tabela MOBILA');
    end if;
END;
/

--Declansarea la:
--insert
INSERT INTO MOBILA(mobila_id, nume, pret_fabrica, data_fabricarii, garantie, sectie_id)
VALUES (6, 'MASA', 2500, TO_DATE('20-MAR-2010', 'dd-MON-yyyy'), 2, 1);

--update
update MOBILA
set pret_fabrica = 1800
where mobila_id = 6;

--delete
delete from MOBILA where mobila_id = 6;
```

```
User-ul GRUPA233 a facut insert in tabela MOBILA


1 row inserted.

User-ul GRUPA233 a facut update in tabela MOBILA


1 row updated.

User-ul GRUPA233 a sters in tabela MOBILA


1 row deleted.
```

11. Am creat un trigger LMD la nivel de linie care nu permite modificarea etajului sectiei din tabela SECTIE.

```
CREATE OR REPLACE TRIGGER ex11
    before update of etaj on SECTIE
    FOR EACH ROW
    WHEN (NEW.etaj <> OLD.etaj)
BEGIN
    raise_application_error(-20000, 'Nu se poate schimba etajul sectiei!');
END;
/

--Declansare:
update sectie
set etaj = 10
where sectie_id = 1;
```

Script Output | Query Result

Task completed in 0.585 seconds

```
Trigger EX11 compiled


Error starting at line : 311 in command -
update sectie
set etaj = 10
where sectie_id = 1
Error report -
ORA-20000: Nu se poate schimba etajul sectiei!
```

12. Am creat un trigger LDD care adauga intr-un tabel toate evenimentele de tip Create, Alter si Drop table.

```
create table audit_schema(
    utilizator VARCHAR2(30),
    nume_bd VARCHAR2(50),
    eveniment VARCHAR2(20),
    nume_obiect VARCHAR2(30),
    data_audit DATE
);

CREATE OR REPLACE TRIGGER ex12
    after create or drop or alter on SCHEMA
BEGIN
    DBMS_OUTPUT.enable;
    DBMS_OUTPUT.PUT_LINE('Trigger declansat');
    INSERT INTO audit_schema
    VALUES (SYS.LOGIN_USER, SYS.DATABASE_NAME, SYS.SYSEVENT, SYS.DICTIONARY_OBJ_NAME, SYSDATE);
END;
/

--Declansarea:
create table tabel(linii number);
alter table tabel add (linii2 number);
drop table tabel;
```

Script Output | Query Result

Task completed in 0.444 seconds

```
Table AUDIT_SCHEMA created.


Trigger EX12 compiled
```

```
Trigger declansat


Table TABEL created.

Trigger declansat


Table TABEL altered.

Trigger declansat


Table TABEL dropped.
```