Gestionarea unui club sportiv

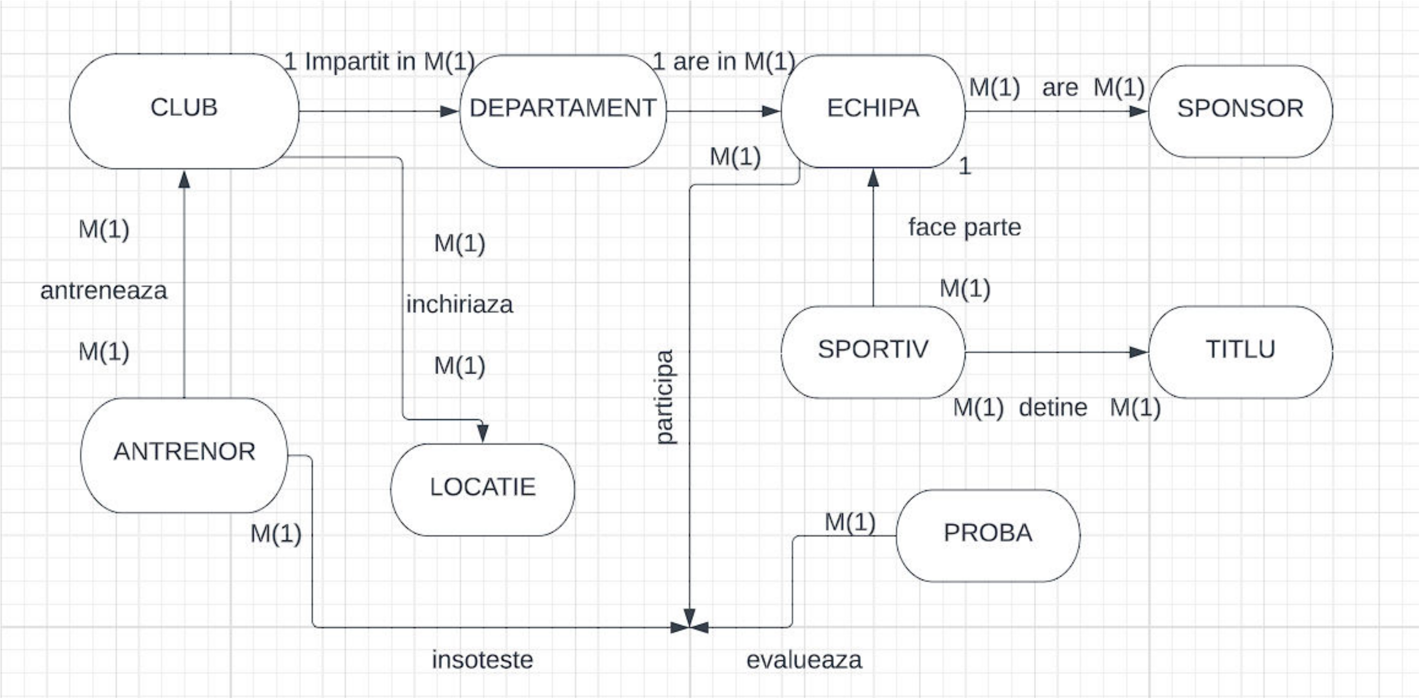
Hurloi Selena-Andreea

Grupa 242

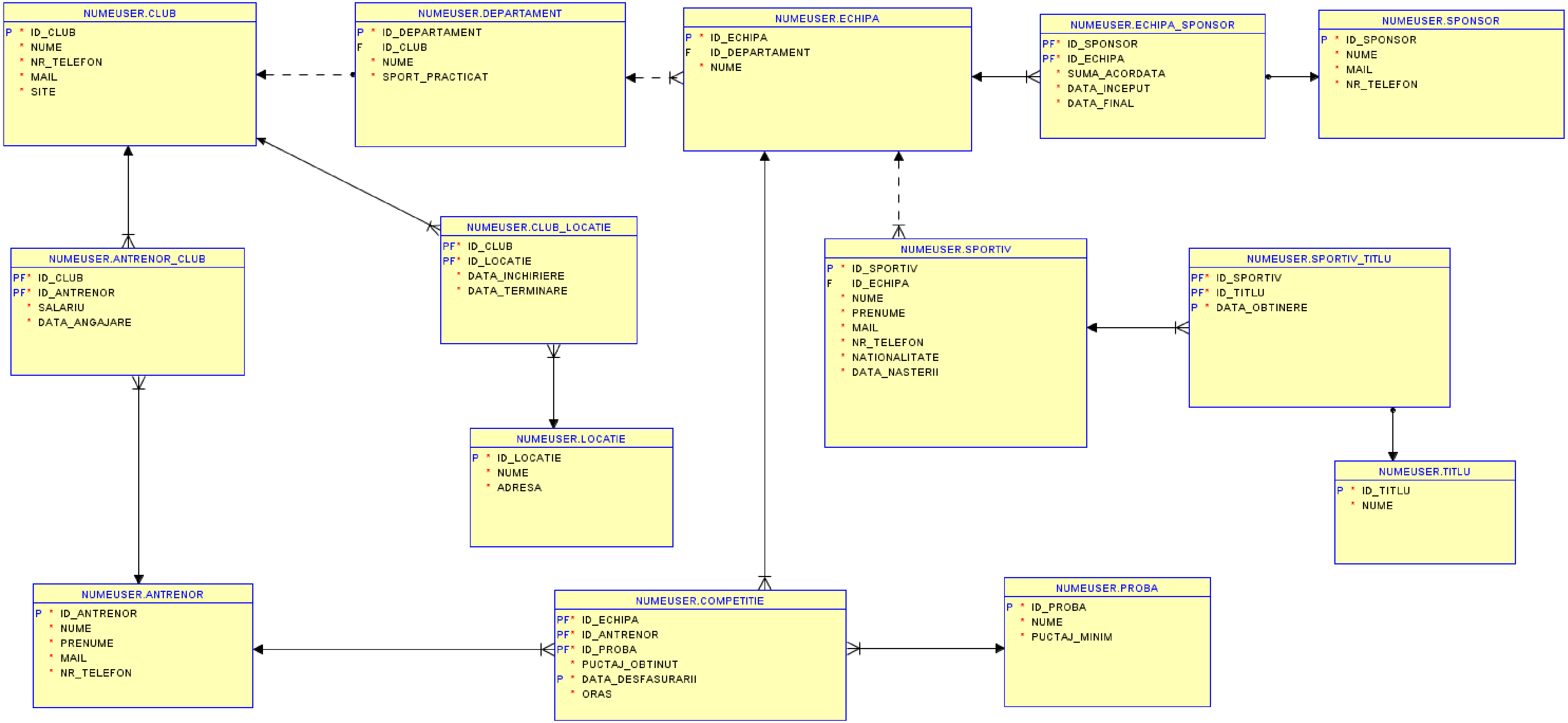
1. Prezentați pe scurt baza de date (utilitatea ei)

Scopul proiectului este de a tine evidenta informatiilor specifice unui club sportiv. Cluburile sportive se impart in mai multe departamente, iar in cadrul departamentelor se gasesc echipele. Aceste echipe pot beneficia de sponsori. Antrenorii lucreaza pentru cluburi, insa acestia se pot angaja la mai multe. Cluburile inchiriaza locatii pentru antrenamente unde sportivii isi desfasoara pregatirea. La competitii, echipele sunt insotite de antrenor, iar in cadrul acestora primesc punctaje. Pentru premiile obtinute in decursul carierei, sportivii sunt premiati cu titluri.

2. Diagrama Entitate-Relație(ERD)

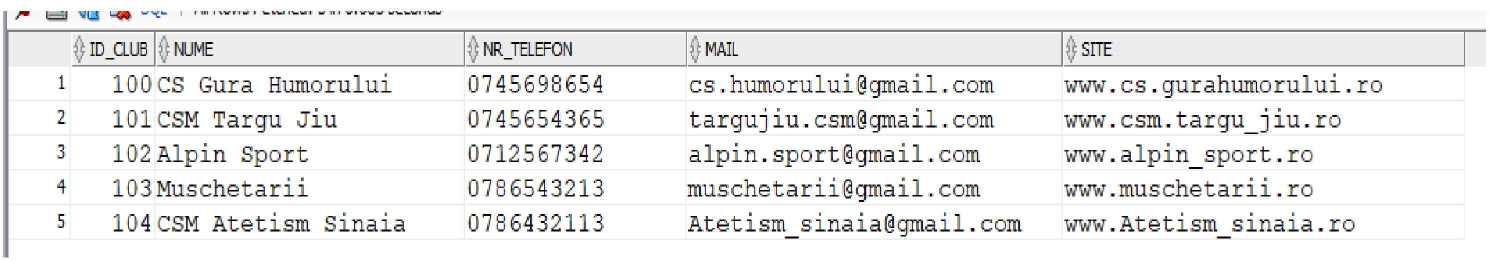


3. Diagrama Conceptuală



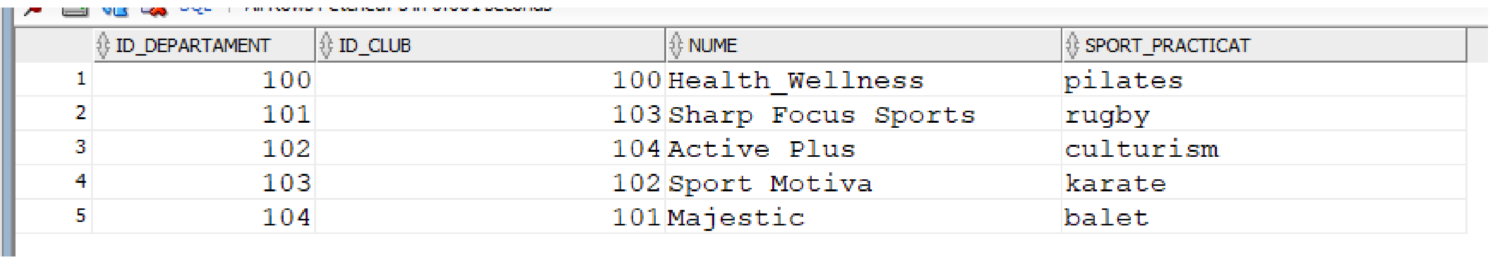
4. Implementare si inserare

```
Create table CLUB
                    (id_club int constraint id_club_pk PRIMARY KEY,
                     nume VARCHAR(20) CONSTRAINT name_club not null,
                     nr_telefon VARCHAR(20) CONSTRAINT nr_telefon_club not
null,
                     mail VARCHAR(200) CONSTRAINT mail_club not null,
                     site VARCHAR (200) CONSTRAINT site_facultate not null
                     );


CREATE SEQUENCE ID
INCREMENT by 1
START WITH 100
MAXVALUE 10000
NOCYCLE;

Insert into Club
values (ID.nextval,'CS Gura
Humorului','0745698654','cs.humorului@gmail.com','www.cs.gurahumorului.ro'
);
Insert into Club
values (ID.nextval,'CSM Targu
Jiu','0745654365','targujiu.csm@gmail.com','www.csm.targu_jiu.ro');
Insert into Club
values (ID.nextval,'Alpin
Sport','0712567342','alpin.sport@gmail.com','www.alpin_sport.ro');
Insert into Club
values
(ID.nextval,'Muschetarii','0786543213','muschetarii@gmail.com','www.musche
tarii.ro');
Insert into Club
values (ID.nextval,'CSM Atetism
Sinaia','0786432113','atetism_sinaia@gmail.com','www.Atetism_sinaia.ro');

select * from club
```



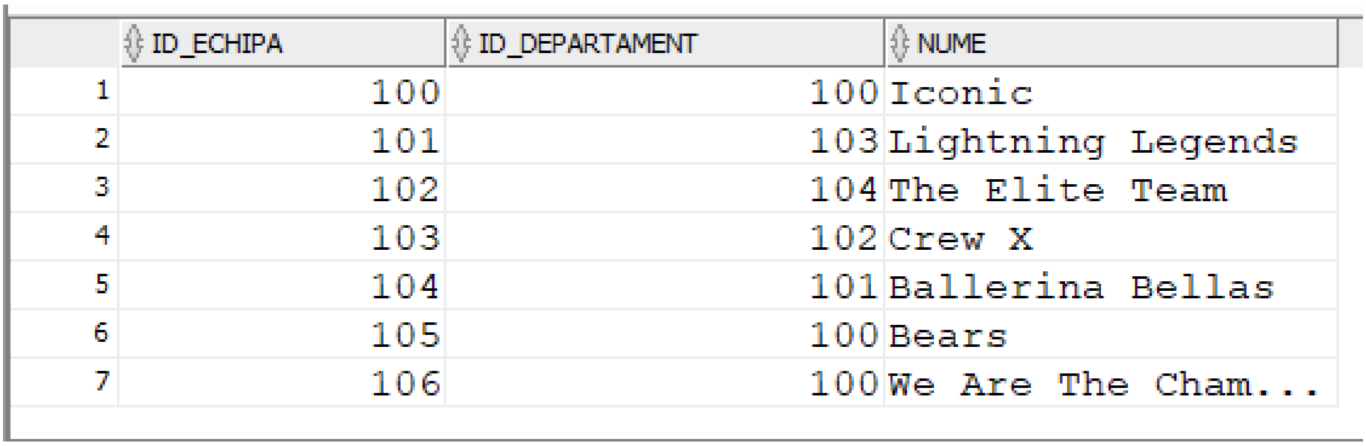
	ID_CLUB	NUME	NR_TELEFON	MAIL	SITE
1	100	CS Gura Humorului	0745698654	cs.humorului@gmail.com	www.cs.gurahumorului.ro
2	101	CSM Targu Jiu	0745654365	targujiu.csm@gmail.com	www.csm.targu_jiu.ro
3	102	Alpin Sport	0712567342	alpin.sport@gmail.com	www.alpin_sport.ro
4	103	Muschetarii	0786543213	muschetarii@gmail.com	www.muschetarii.ro
5	104	CSM Atetism Sinaia	0786432113	Atetism_sinaia@gmail.com	www.Atetism_sinaia.ro

```
create table DEPARTAMENT
                 (id_departament int  constraint id_departament_pk PRIMARY
KEY,
                  id_club int,  constraint id_club_fk FOREIGN KEY
(id_club) references CLUB(id_club),
                  nume varchar (20) not null,
                  sport_practicat varchar (20) not null
                  );


CREATE SEQUENCE ID_DEP
INCREMENT BY 1
START WITH 100
MAXVALUE 100000
NOCYCLE;

INSERT INTO Departament
VALUES (ID_DEP.NEXTVAL,100,'Health_Wellness','pilates');
INSERT INTO Departament
VALUES (ID_DEP.NEXTVAL,103,'Sharp Focus Sports','rugby');
INSERT INTO Departament
VALUES (ID_DEP.NEXTVAL,104,'Active Plus','culturism');
INSERT INTO Departament
VALUES (ID_DEP.NEXTVAL,102,'Sport Motiva','karate');
INSERT INTO Departament
VALUES (ID_DEP.NEXTVAL,101,'Majestic','balet');

               SELECT * FROM departament
```



	ID_DEPARTAMENT	ID_CLUB	NUME	SPORT_PRACTICAT
1	100	100	Health_Wellness	pilates
2	101	103	Sharp Focus Sports	rugby
3	102	104	Active Plus	culturism
4	103	102	Sport Motiva	karate
5	104	101	Majestic	balet

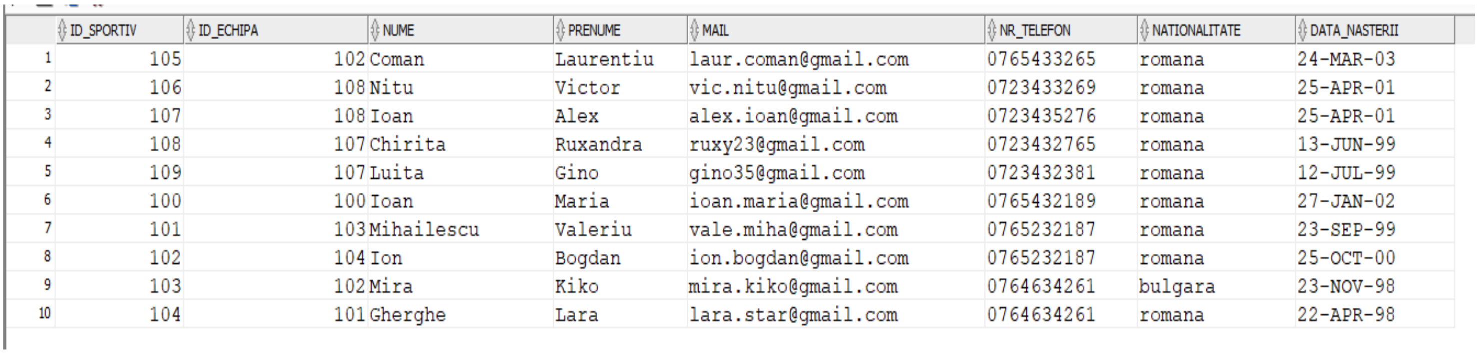
```
CREATE TABLE ECHIPA (id_echipa int constraint id_echila_pk PRIMARY
KEY,
                    id_departament int,  constraint
id_departament_fk FOREIGN KEY (id_departament) references
DEPARTAMENT(id_departament),
                    nume VARCHAR(20) NOT NULL
                    );
CREATE SEQUENCE ID_ECH
INCREMENT BY 1
START WITH 100
MAXVALUE 100000
NOCYCLE;

INSERT INTO Echipa
VALUES (ID_ECH.NEXTVAL,100,'Iconic');
INSERT INTO Echipa
VALUES (ID_ECH.NEXTVAL,103,'Lightning Legends');
INSERT INTO Echipa
VALUES (ID_ECH.NEXTVAL,104,'The Elite Team');
INSERT INTO Echipa
VALUES (ID_ECH.NEXTVAL,102,'Crew X');
INSERT INTO Echipa
VALUES (ID_ECH.NEXTVAL,101,'Ballerina Bellas');
INSERT INTO Echipa
VALUES (105,100,'Bears');
INSERT INTO Echipa
VALUES (106,100,'We Are The Champions');

SELECT * FROM echipa
```



	ID_ECHIPA	ID_DEPARTAMENT	NUME
1	100	100	Iconic
2	101	103	Lightning Legends
3	102	104	The Elite Team
4	103	102	Crew X
5	104	101	Ballerina Bellas
6	105	100	Bears
7	106	100	We Are The Cham...

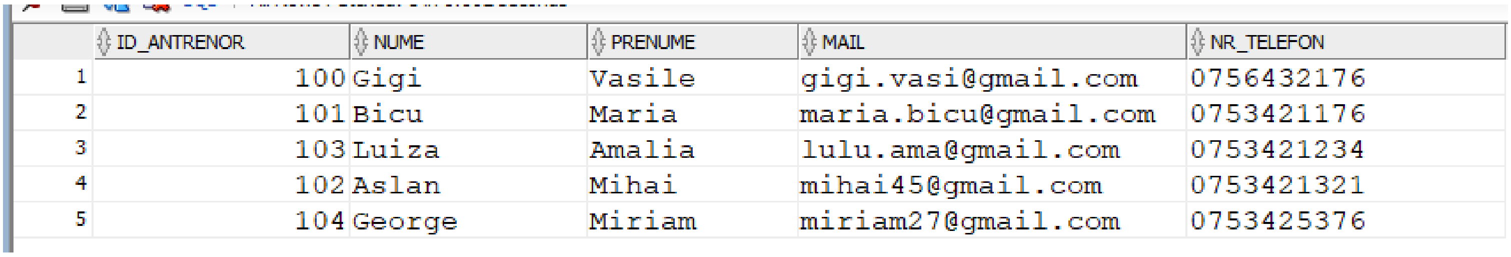
```
CREATE TABLE SPORTIV ( id_sportiv int  constraint id_sportiv_pk PRIMARY
KEY,
                       id_echipa int,  constraint id_echipa_fk FOREIGN KEY
(id_echipa) references ECHIPA(id_echipa),
                       nume VARCHAR(20) CONSTRAINT name_sportiv NOT NULL,
                       prenume VARCHAR(20) CONSTRAINT prenume_sportiv NOT
NULL,
                       mail VARCHAR(20) CONSTRAINT mail_sportiv NOT NULL,
                       nr_telefon VARCHAR(20) NOT NULL,
                       nationalitate VARCHAR(20) NOT NULL,
                       data_nasterii DATE NOT NULL
                       );
CREATE SEQUENCE ID_sportiv2
INCREMENT BY 1
START WITH 100
MAXVALUE 100000
NOCYCLE;

INSERT INTO Sportiv
VALUES(ID_sportiv2.NEXTVAL, 100, 'Ioan', 'Maria',
'ioan.maria@gmail.com','0765432189','romana',
TO_DATE('27-01-2002','dd-mm-yyyy'));
INSERT INTO SPORTIV
VALUES(ID_sportiv2.NEXTVAL, 103, 'Mihailescu', 'Valeriu',
'vale.miha@gmail.com','0765232187','romana',
TO_DATE('23-09-1999','dd-mm-yyyy'));
INSERT INTO SPORTIV
VALUES(ID_SPORTIV2.NEXTVAL, 104, 'Ion', 'Bogdan',
'ion.bogdan@gmail.com','0765232187','romana',
TO_DATE('25-10-2000','dd-mm-yyyy'));
INSERT INTO SPORTIV
VALUES(ID_SPORTIV2.NEXTVAL, 102, 'Mira', 'Kiko',
'mira.kiko@gmail.com','0764634261','bulgara',
TO_DATE('23-11-1998','dd-mm-yyyy'));
INSERT INTO SPORTIV
VALUES(ID_SPORTIV2.NEXTVAL, 101, 'Gherghe', 'Lara',
'lara.star@gmail.com','0764634261','romana',
TO_DATE('22-04-1998','dd-mm-yyyy'));
INSERT INTO Sportiv
VALUES(105, 102, 'Coman', 'Laurentiu',
'laur.coman@gmail.com','0765433265','romana',
TO_DATE('24-03-2003','dd-mm-yyyy'));
INSERT INTO Sportiv
VALUES(106, 108, 'Nitu', 'Victor',
'vic.nitu@gmail.com','0723433269','romana',
TO_DATE('25-04-2001','dd-mm-yyyy'));
INSERT INTO Sportiv
VALUES(107, 108, 'Ioan', 'Alex',
'alex.ioan@gmail.com','0723435276','romana',
TO_DATE('25-04-2001','dd-mm-yyyy'));
```

```
INSERT INTO Sportiv
VALUES(108, 107, 'Chirita', 'Ruxandra',
'ruxy23@gmail.com','0723432765','romana',
TO_DATE('13-06-1999','dd-mm-yyyy'));
INSERT INTO Sportiv
VALUES(109, 107, 'Luita', 'Gino',
'gino35@gmail.com','0723432381','romana',
TO_DATE('12-07-1999','dd-mm-yyyy'));

SELECT * FROM sportiv
```



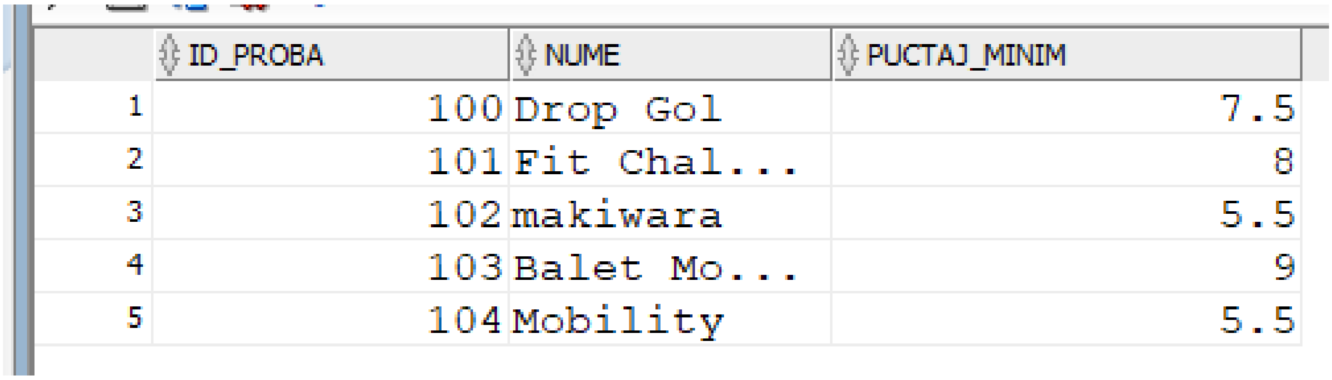
	ID_SPORTIV	ID_ECHIPA	NUME	PRENUME	MAIL	NR_TELEFON	NATIONALITATE	DATA_NASTERII
1	105	102	Coman	Laurentiu	laur.coman@gmail.com	0765433265	romana	24-MAR-03
2	106	108	Nitu	Victor	vic.nitu@gmail.com	0723433269	romana	25-APR-01
3	107	108	Ioan	Alex	alex.ioan@gmail.com	0723435276	romana	25-APR-01
4	108	107	Chirita	Ruxandra	ruxy23@gmail.com	0723432765	romana	13-JUN-99
5	109	107	Luita	Gino	gino35@gmail.com	0723432381	romana	12-JUL-99
6	100	100	Ioan	Maria	ioan.maria@gmail.com	0765432189	romana	27-JAN-02
7	101	103	Mihailescu	Valeriu	vale.miha@gmail.com	0765232187	romana	23-SEP-99
8	102	104	Ion	Bogdan	ion.bogdan@gmail.com	0765232187	romana	25-OCT-00
9	103	102	Mira	Kiko	mira.kiko@gmail.com	0764634261	bulgara	23-NOV-98
10	104	101	Gherghe	Lara	lara.star@gmail.com	0764634261	romana	22-APR-98

```
CREATE TABLE ANTRENOR (id_antrenor int  constraint id_antrenor_pk PRIMARY
KEY,
                       nume VARCHAR(20) CONSTRAINT name_antrenor NOT NULL,
                       prenume VARCHAR(20) CONSTRAINT prenume_antrenor NOT
NULL,
                       mail VARCHAR(20) CONSTRAINT mail_antrenor NOT NULL,
                       nr_telefon VARCHAR(20) NOT NULL);
CREATE SEQUENCE ID_antrenor
INCREMENT BY 1
START WITH 100
MAXVALUE 100000
NOCYCLE;

INSERT INTO ANTRENOR
VALUES
(ID_antrenor.NEXTVAL,'Gigi','Vasile','gigi.vasi@gmail.com','0756432176');
INSERT INTO ANTRENOR
VALUES
(ID_antrenor.NEXTVAL,'Bicu','Maria','maria.bicu@gmail.com','0753421176');
INSERT INTO ANTRENOR
VALUES
(ID_antrenor.NEXTVAL,'Luiza','Amalia','lulu.ama@gmail.com','0753421234');
INSERT INTO ANTRENOR
VALUES (102,'Aslan','Mihai','mihai45@gmail.com','0753421321');
INSERT INTO ANTRENOR
VALUES
(ID_antrenor.NEXTVAL,'George','Miriam','miriam27@gmail.com','0753425376');

SELECT * FROM antrenor
```



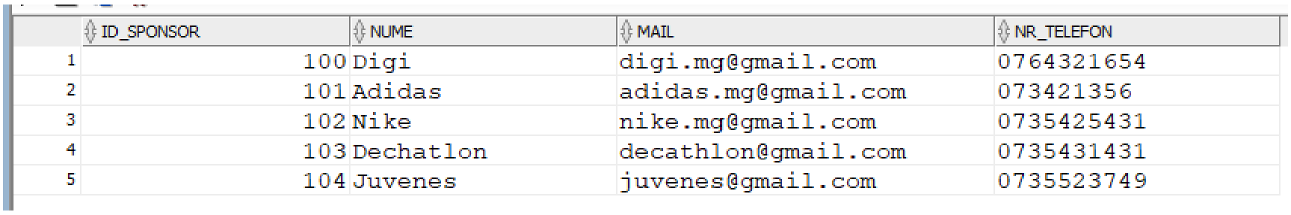
	ID_ANTRENOR	NUME	PRENUME	MAIL	NR_TELEFON
1	100	Gigi	Vasile	gigi.vasi@gmail.com	0756432176
2	101	Bicu	Maria	maria.bicu@gmail.com	0753421176
3	103	Luiza	Amalia	lulu.ama@gmail.com	0753421234
4	102	Aslan	Mihai	mihai45@gmail.com	0753421321
5	104	George	Miriam	miriam27@gmail.com	0753425376

```
CREATE TABLE PROBA  (id_proba int  constraint id_proba_pk PRIMARY KEY,
                     nume VARCHAR(20) CONSTRAINT name_proba NOT NULL,
                     puctaj_minim NUMBER(7,2) NOT NULL);

CREATE SEQUENCE id_proba
INCREMENT BY 1
START WITH 100
MAXVALUE 100000
NOCYCLE;

INSERT INTO proba
VALUES(id_proba.NEXTVAL,'Drop Gol',7.5)
INSERT INTO proba
VALUES(id_proba.NEXTVAL,'Fit Challenge',8)
INSERT INTO proba
VALUES(id_proba.NEXTVAL,'makiwara',5.5)
INSERT INTO proba
VALUES(id_proba.NEXTVAL,'Balet Modobradi',9)
INSERT INTO proba
VALUES(id_proba.NEXTVAL,'Mobility',5.5)

SELECT * FROM proba
```



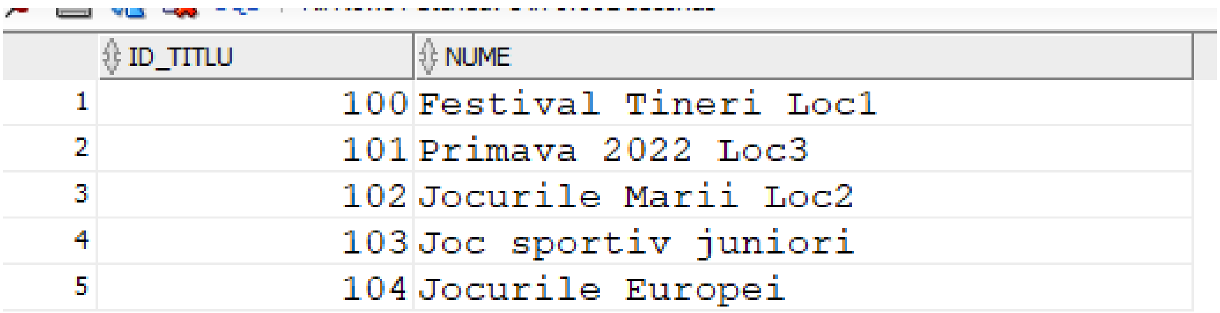
	ID_PROBA	NUME	PUCTAJ_MINIM
1	100	Drop Gol	7.5
2	101	Fit Chal...	8
3	102	makiwara	5.5
4	103	Balet Mo...	9
5	104	Mobility	5.5

```
CREATE TABLE SPONSOR (id_sponsor int  constraint id_sponsor_pk PRIMARY
KEY,
                      nume VARCHAR(20) CONSTRAINT name_sponsor NOT NULL,
                      mail VARCHAR(20) CONSTRAINT mail_sponsor NOT NULL,
                      nr_telefon VARCHAR(20) NOT NULL);
CREATE SEQUENCE id_sponsor
INCREMENT BY 1
START WITH 100
MAXVALUE 100000
NOCYCLE;

INSERT INTO sponsor
VALUES(id_sponsor.NEXTVAL,'Digi','digi.mg@gmail.com','0764321654')
INSERT INTO sponsor
VALUES(id_sponsor.NEXTVAL,'Adidas','adidas.mg@gmail.com','073421356')
INSERT INTO sponsor
VALUES(id_sponsor.NEXTVAL,'Nike','nike.mg@gmail.com','0735425431')
INSERT INTO sponsor
VALUES(id_sponsor.NEXTVAL,'Dechatlon','decathlon@gmail.com','0735431431')
INSERT INTO sponsor
VALUES(id_sponsor.NEXTVAL,'Juvenes','juvenes@gmail.com','0735523749')

SELECT * FROM sponsor
```



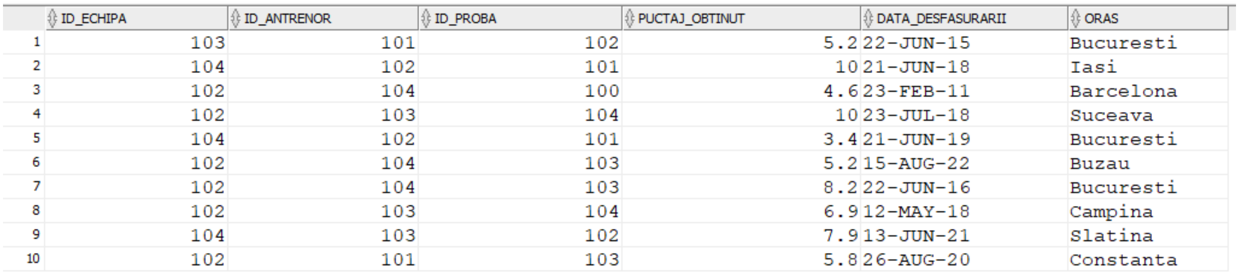
	ID_SPONSOR	NUME	MAIL	NR_TELEFON
1	100	Digi	digi.mg@gmail.com	0764321654
2	101	Adidas	adidas.mg@gmail.com	073421356
3	102	Nike	nike.mg@gmail.com	0735425431
4	103	Dechatlon	decathlon@gmail.com	0735431431
5	104	Juvenes	juvenes@gmail.com	0735523749

```
CREATE TABLE TITLU (id_titlu int  constraint id_titlu_pk PRIMARY KEY,
                    nume VARCHAR(20) CONSTRAINT name_titlu NOT NULL);
CREATE SEQUENCE id_titlu
INCREMENT BY 1
START WITH 100
MAXVALUE 100000
NOCYCLE;

INSERT INTO titlu
VALUES(id_titlu.NEXTVAL,'Festival Tineri Loc1')
INSERT INTO titlu
VALUES(id_titlu.NEXTVAL,'Primava 2022 Loc3')
INSERT INTO titlu
VALUES(id_titlu.NEXTVAL,'Jocurile Marii Loc2')
INSERT INTO titlu
VALUES(id_titlu.NEXTVAL,'Joc sportiv juniori')
INSERT INTO titlu
VALUES(id_titlu.NEXTVAL,'Jocurile Europei')

SELECT * FROM titlu
```



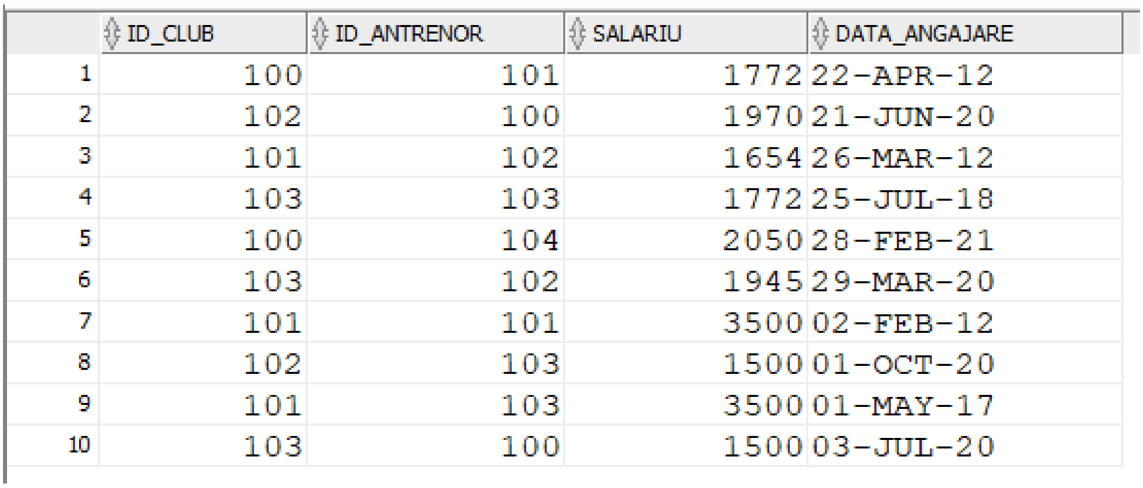
	ID_TITLU	NUME
1	100	Festival Tineri Loc1
2	101	Primava 2022 Loc3
3	102	Jocurile Marii Loc2
4	103	Joc sportiv juniori
5	104	Jocurile Europei

```
CREATE TABLE COMPETITIE (id_echipa int,  constraint id_echipa_fk_comp
FOREIGN KEY (id_echipa) references ECHIPA(id_echipa),
                          id_antrenor int,  constraint id_antrenor_fk_comp
FOREIGN KEY (id_antrenor) references ANTRENOR(id_antrenor),
                          id_proba int,  constraint id_proba_fk_comp
FOREIGN KEY (id_proba) references PROBA(id_proba),
                          puctaj_obtinut NUMBER(7,2) NOT NULL,
                          data_desfasurarii DATE NOT NULL,
                          oras VARCHAR(20) NOT NULL,
                          CONSTRAINT pk_competitie primary
key(id_echipa,id_antrenor,id_proba)
                          );
INSERT INTO competitie
VALUES(103,101,102,5.2,TO_DATE('22-06-2015','dd-mm-yyyy'),'Bucuresti')
INSERT INTO competitie
VALUES(104,102,101,10,TO_DATE('21-06-2018','dd-mm-yyyy'),'Iasi')
INSERT INTO competitie
VALUES(102,104,100,4.6,TO_DATE('23-02-2011','dd-mm-yyyy'),'Barcelona')
INSERT INTO competitie
VALUES(102,103,104,10,TO_DATE('23-07-2018','dd-mm-yyyy'),'Suceava')
INSERT INTO competitie
VALUES(104,102,101,3.4,TO_DATE('21-06-2019','dd-mm-yyyy'),'Bucuresti')
INSERT INTO competitie
VALUES(102,104,103,5.2,TO_DATE('15-08-2022','dd-mm-yyyy'),'Buzau')
INSERT INTO competitie
VALUES(102,104,103,8.2,TO_DATE('22-06-2016','dd-mm-yyyy'),'Bucuresti')
INSERT INTO competitie
VALUES(102,103,104,6.9,TO_DATE('12-05-2018','dd-mm-yyyy'),'Campina')
INSERT INTO competitie
VALUES(104,103,102,7.9,TO_DATE('13-06-2021','dd-mm-yyyy'),'Slatina')
INSERT INTO competitie
VALUES(102,101,103,5.8,TO_DATE('26-08-2020','dd-mm-yyyy'),'Constanta')
INSERT INTO competitie
VALUES(108,101,102,10,TO_DATE('22-06-2015','dd-mm-yyyy'),'Bucuresti')
INSERT INTO competitie
VALUES(107,102,104,3,TO_DATE('22-06-2015','dd-mm-yyyy'),'Galati')
```



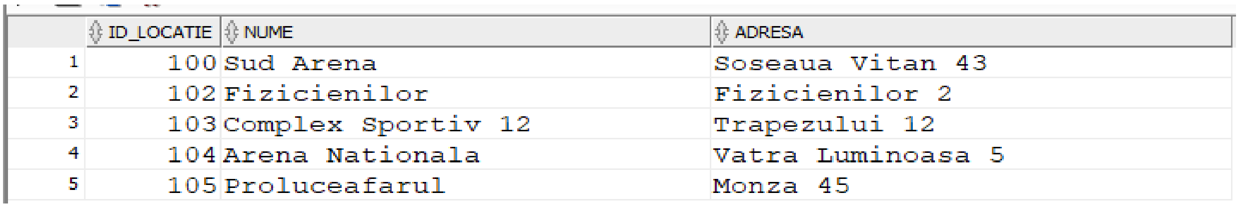
	ID_ECHIPA	ID_ANTRENOR	ID_PROBA	PUCTAJ_OBTINUT	DATA_DESFASURARII	ORAS
1	103	101	102	5.2	22-JUN-15	Bucuresti
2	104	102	101	10	21-JUN-18	Iasi
3	102	104	100	4.6	23-FEB-11	Barcelona
4	102	103	104	10	23-JUL-18	Suceava
5	104	102	101	3.4	21-JUN-19	Bucuresti
6	102	104	103	5.2	15-AUG-22	Buzau
7	102	104	103	8.2	22-JUN-16	Bucuresti
8	102	103	104	6.9	12-MAY-18	Campina
9	104	103	102	7.9	13-JUN-21	Slatina
10	102	101	103	5.8	26-AUG-20	Constanta

```
CREATE TABLE ANTRENOR_CLUB (id_club int,  constraint id_club_fk_comp
FOREIGN KEY (id_club) references CLUB(id_club),
                            id_antrenor int,  constraint
id_antrenor_fk_compus FOREIGN KEY (id_antrenor) references
ANTRENOR(id_antrenor),
                            salariu int NOT NULL,
                            data_angajare DATE NOT NULL,
                            CONSTRAINT pk_antrenor_club primary
key(id_club,id_antrenor)
                            );
INSERT INTO Antrenor_Club
VALUES(100,101,1772,TO_DATE('22-04-2012','dd-mm-yyyy'))
INSERT INTO Antrenor_Club
VALUES(102,100,1970,TO_DATE('21-06-2020','dd-mm-yyyy'))
INSERT INTO Antrenor_Club
VALUES(101,102,1654,TO_DATE('26-03-2012','dd-mm-yyyy'))
INSERT INTO Antrenor_Club
VALUES(103,103,1772,TO_DATE('25-07-2018','dd-mm-yyyy'))
INSERT INTO Antrenor_Club
VALUES(100,104,2050,TO_DATE('28-02-2021','dd-mm-yyyy'))
INSERT INTO Antrenor_Club
VALUES(103,102,1945,TO_DATE('29-03-2020','dd-mm-yyyy'))
INSERT INTO Antrenor_Club
VALUES(101,101,3500,TO_DATE('02-02-2012','dd-mm-yyyy'))
INSERT INTO Antrenor_Club
VALUES(102,103,1500,TO_DATE('01-10-2020','dd-mm-yyyy'))
INSERT INTO Antrenor_Club
VALUES(101,103,3500,TO_DATE('01-05-2017','dd-mm-yyyy'))
INSERT INTO Antrenor_Club
VALUES(103,100,1500,TO_DATE('03-07-2020','dd-mm-yyyy'))

SELECT * FROM antrenor_club
```



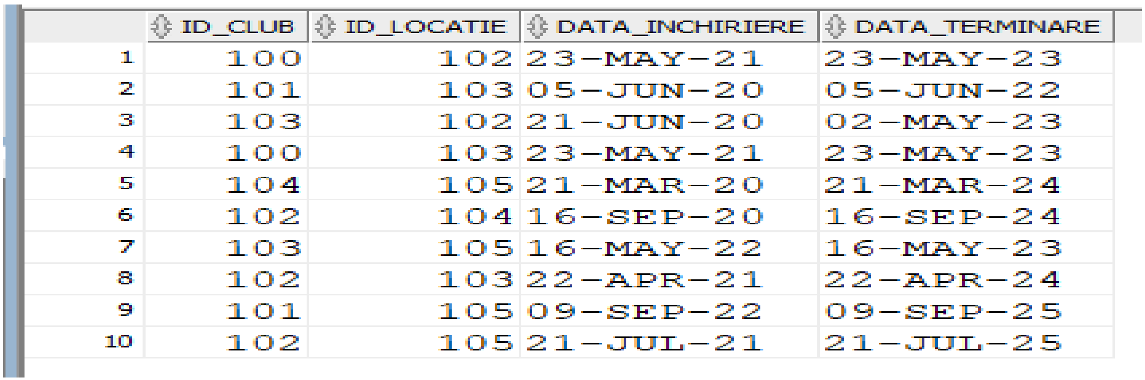
	ID_CLUB	ID_ANTRENOR	SALARIU	DATA_ANGAJARE
1	100	101	1772	22-APR-12
2	102	100	1970	21-JUN-20
3	101	102	1654	26-MAR-12
4	103	103	1772	25-JUL-18
5	100	104	2050	28-FEB-21
6	103	102	1945	29-MAR-20
7	101	101	3500	02-FEB-12
8	102	103	1500	01-OCT-20
9	101	103	3500	01-MAY-17
10	103	100	1500	03-JUL-20

```
CREATE TABLE LOCATIE (id_locatie int  constraint id_locatie_pk PRIMARY
KEY,
                      nume VARCHAR(20) CONSTRAINT name_locatie NOT NULL,
                      adresa VARCHAR(20) CONSTRAINT adresa_locatie NOT
NULL
                      );
CREATE SEQUENCE id_locatie
INCREMENT BY 1
START WITH 100
MAXVALUE 100000
NOCYCLE;

INSERT INTO locatie
VALUES(id_locatie.NEXTVAL,'Sud Arena','Soseaua Vitan 43')
INSERT INTO locatie
VALUES(id_locatie.NEXTVAL,'Fizicienilor','Fizicienilor 2')
INSERT INTO locatie
VALUES(id_locatie.NEXTVAL,'Complex Sportiv 12','Trapezului 12')
INSERT INTO locatie
VALUES(id_locatie.NEXTVAL,'Arena Nationala','Vatra Luminoasa 5')
INSERT INTO locatie
VALUES(id_locatie.NEXTVAL,'Proluceafarul','Monza 45')

SELECT * FROM locatie
```



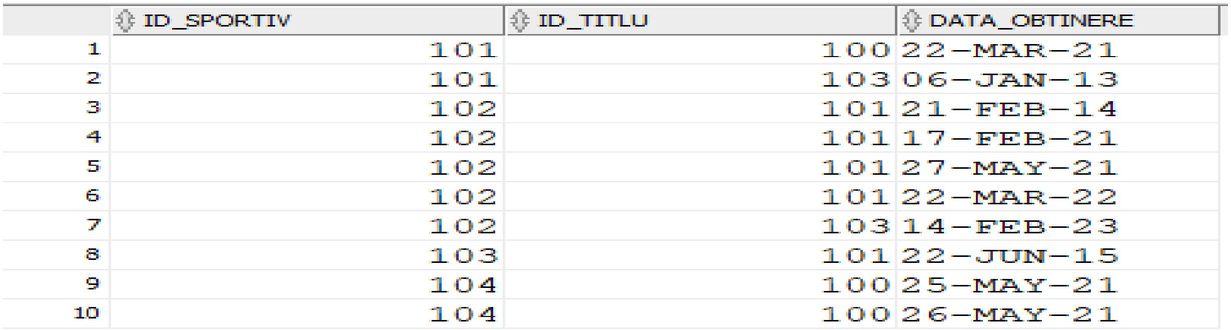
	ID_LOCATIE	NUME	ADRESA
1	100	Sud Arena	Soseaua Vitan 43
2	102	Fizicienilor	Fizicienilor 2
3	103	Complex Sportiv 12	Trapezului 12
4	104	Arena Nationala	Vatra Luminoasa 5
5	105	Proluceafarul	Monza 45

```
CREATE TABLE CLUB_LOCATIE (id_club int,  constraint id_club_fk_comp_loc
FOREIGN KEY (id_club) references CLUB(id_club),
                          id_locatie int,  constraint id_loc_fk_comp
FOREIGN KEY (id_locatie) references LOCATIE(id_locatie),
                          data_inchiriere DATE NOT NULL,
                          data_terminare DATE NOT NULL,
                          CONSTRAINT pk_club_locatie primary
key(id_club,id_locatie)
                          );
INSERT INTO Club_Locatie
VALUES(100,102,TO_DATE('23-05-2021','dd-mm-yyyy'),TO_DATE('23-05-
2023','dd-mm-yyyy'))
INSERT INTO Club_Locatie
VALUES(101,103,TO_DATE('05-06-2020','dd-mm-yyyy'),TO_DATE('05-06-
2022','dd-mm-yyyy'))
INSERT INTO Club_Locatie
VALUES(103,102,TO_DATE('21-06-2020','dd-mm-yyyy'),TO_DATE('2-05-2023','dd-
mm-yyyy'))
INSERT INTO Club_Locatie
VALUES(100,103,TO_DATE('23-05-2021','dd-mm-yyyy'),TO_DATE('23-05-
2023','dd-mm-yyyy'))
INSERT INTO Club_Locatie
VALUES(104,105,TO_DATE('21-03-2020','dd-mm-yyyy'),TO_DATE('21-03-
2024','dd-mm-yyyy'))
INSERT INTO Club_Locatie
VALUES(102,104,TO_DATE('16-09-2020','dd-mm-yyyy'),TO_DATE('16-09-
2024','dd-mm-yyyy'))
INSERT INTO Club_Locatie
VALUES(100,102,TO_DATE('23-05-2021','dd-mm-yyyy'),TO_DATE('23-05-
2023','dd-mm-yyyy'))
INSERT INTO Club_Locatie
VALUES(103,105,TO_DATE('16-05-2022','dd-mm-yyyy'),TO_DATE('16-05-
2023','dd-mm-yyyy'))
INSERT INTO Club_Locatie
VALUES(102,103,TO_DATE('22-04-2021','dd-mm-yyyy'),TO_DATE('22-04-
2024','dd-mm-yyyy'))
INSERT INTO Club_Locatie
VALUES(101,105,TO_DATE('09-09-2022','dd-mm-yyyy'),TO_DATE('09-09-
2025','dd-mm-yyyy'))
INSERT INTO Club_Locatie
VALUES(102,105,TO_DATE('21-07-2021','dd-mm-yyyy'),TO_DATE('21-07-
2025','dd-mm-yyyy'))

SELECT * FROM club_locatie
```



	ID_CLUB	ID_LOCATIE	DATA_INCHIRIERE	DATA_TERMINARE
1	100	102	23-MAY-21	23-MAY-23
2	101	103	05-JUN-20	05-JUN-22
3	103	102	21-JUN-20	02-MAY-23
4	100	103	23-MAY-21	23-MAY-23
5	104	105	21-MAR-20	21-MAR-24
6	102	104	16-SEP-20	16-SEP-24
7	103	105	16-MAY-22	16-MAY-23
8	102	103	22-APR-21	22-APR-24
9	101	105	09-SEP-22	09-SEP-25
10	102	105	21-JUL-21	21-JUL-25

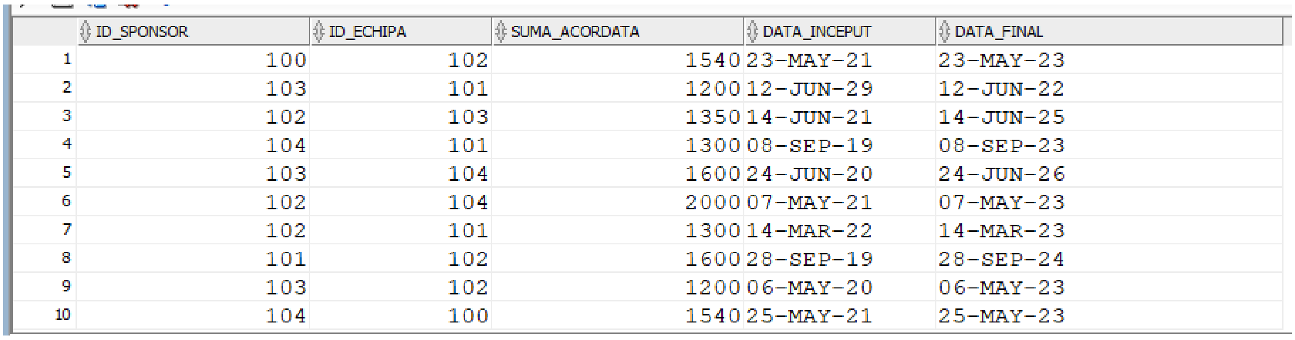
```
CREATE TABLE SPORTIV_TITLU (id_sportiv int,  constraint
id_sportiv_fk_titlu FOREIGN KEY (id_sportiv) references
SPORTIV(id_sportiv),
                               id_titlu int,  constraint id_titlu_fk_titlu
FOREIGN KEY (id_titlu) references TITLU(id_titlu),
                               data_obtinere DATE NOT NULL,
                               CONSTRAINT pk_sportiv_titlu_compus primary
key(id_sportiv,id_titlu)
                            );

INSERT INTO sportiv_titlu
VALUES(104,100,TO_DATE('25-05-2021','dd-mm-yyyy'))
INSERT INTO sportiv_titlu
VALUES(104,100,TO_DATE('26-05-2021','dd-mm-yyyy'))
INSERT INTO sportiv_titlu
VALUES(102,101,TO_DATE('21-02-2014','dd-mm-yyyy'))
INSERT INTO sportiv_titlu
VALUES(103,101,TO_DATE('22-06-2015','dd-mm-yyyy'))
INSERT INTO sportiv_titlu
VALUES(101,103,TO_DATE('06-01-2013','dd-mm-yyyy'))
INSERT INTO sportiv_titlu
VALUES(102,101,TO_DATE('17-02-2021','dd-mm-yyyy'))
INSERT INTO sportiv_titlu
VALUES(102,101,TO_DATE('22-03-2022','dd-mm-yyyy'))
INSERT INTO sportiv_titlu
VALUES(101,100,TO_DATE('22-03-2021','dd-mm-yyyy'))
INSERT INTO sportiv_titlu
VALUES(102,103,TO_DATE('14-02-2023','dd-mm-yyyy'))
INSERT INTO sportiv_titlu
VALUES(102,101,TO_DATE('27-05-2021','dd-mm-yyyy'))

SELECT * FROM sportiv_titlu
```



	ID_SPORTIV	ID_TITLU	DATA_OBTINERE
1	101	100	22-MAR-21
2	101	103	06-JAN-13
3	102	101	21-FEB-14
4	102	101	17-FEB-21
5	102	101	27-MAY-21
6	102	101	22-MAR-22
7	102	103	14-FEB-23
8	103	101	22-JUN-15
9	104	100	25-MAY-21
10	104	100	26-MAY-21

```
CREATE TABLE ECHIPA_SPONSOR (  id_sponsor int,  constraint
id_sponsor_fk_echipa FOREIGN KEY (id_sponsor) references
SPONSOR(id_sponsor),
                               id_echipa int,  constraint
id_echipa_fk_sponsor FOREIGN KEY (id_echipa) references ECHIPA(id_echipa),
                               suma_acordata int NOT NULL,
                               data_inceput DATE NOT NULL,
                               data_final DATE NOT NULL,
                               CONSTRAINT pk_echipa_sponsor_compus primary
key(id_sponsor,id_echipa)
                               );
INSERT INTO Echipa_Sponsor
VALUES(100,102,1540,TO_DATE('23-05-2021','dd-mm-yyyy'),TO_DATE('23-05-
2023','dd-mm-yyyy'))
INSERT INTO Echipa_Sponsor
VALUES(103,101,1200,TO_DATE('12-06-2029','dd-mm-yyyy'),TO_DATE('12-06-
2022','dd-mm-yyyy'))
INSERT INTO Echipa_Sponsor
VALUES(102,103,1350,TO_DATE('14-06-2021','dd-mm-yyyy'),TO_DATE('14-06-
2025','dd-mm-yyyy'))
INSERT INTO Echipa_Sponsor
VALUES(104,101,1300,TO_DATE('08-09-2019','dd-mm-yyyy'),TO_DATE('08-09-
2023','dd-mm-yyyy'))
INSERT INTO Echipa_Sponsor
VALUES(103,104,1600,TO_DATE('24-06-2020','dd-mm-yyyy'),TO_DATE('24-06-
2026','dd-mm-yyyy'))
INSERT INTO Echipa_Sponsor
VALUES(102,104,2000,TO_DATE('07-05-2021','dd-mm-yyyy'),TO_DATE('07-05-
2023','dd-mm-yyyy'))
INSERT INTO Echipa_Sponsor
VALUES(102,101,1300,TO_DATE('14-03-2022','dd-mm-yyyy'),TO_DATE('14-03-
2023','dd-mm-yyyy'))
INSERT INTO Echipa_Sponsor
VALUES(101,102,1600,TO_DATE('28-09-2019','dd-mm-yyyy'),TO_DATE('28-09-
2024','dd-mm-yyyy'))
INSERT INTO Echipa_Sponsor
VALUES(103,102,1200,TO_DATE('06-05-2020','dd-mm-yyyy'),TO_DATE('06-05-
2023','dd-mm-yyyy'))
```

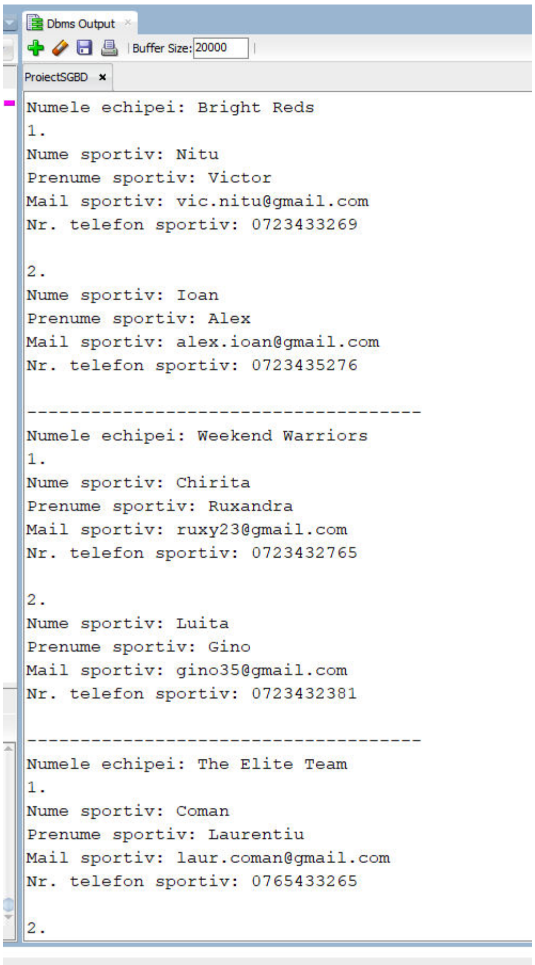
```
INSERT INTO Echipa_Sponsor
VALUES(104,100,1540,TO_DATE('25-05-2021','dd-mm-yyyy'),TO_DATE('25-05-
2023','dd-mm-yyyy'))

SELECT * FROM echipa_sponsor
```



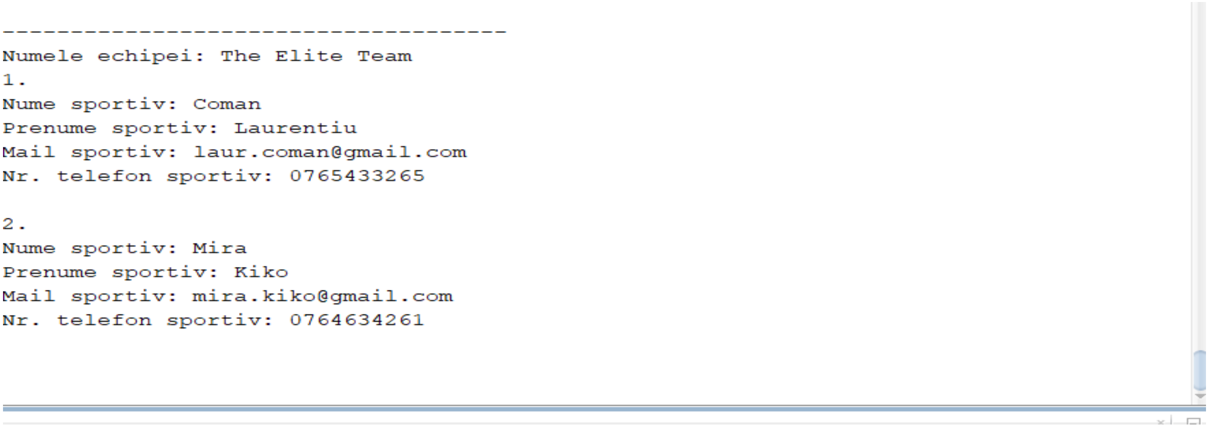
	ID_SPONSOR	ID_ECHIPA	SUMA_ACORDATA	DATA_INCEPUT	DATA_FINAL
1	100	102	1540	23-MAY-21	23-MAY-23
2	103	101	1200	12-JUN-29	12-JUN-22
3	102	103	1350	14-JUN-21	14-JUN-25
4	104	101	1300	08-SEP-19	08-SEP-23
5	103	104	1600	24-JUN-20	24-JUN-26
6	102	104	2000	07-MAY-21	07-MAY-23
7	102	101	1300	14-MAR-22	14-MAR-23
8	101	102	1600	28-SEP-19	28-SEP-24
9	103	102	1200	06-MAY-20	06-MAY-23
10	104	100	1540	25-MAY-21	25-MAY-23

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.

```
Pentru un Club dat, afisati toti sportivii(nume si DATE de contact) din
echipele din cadrul clubului.

CREATE OR REPLACE PROCEDURE cerinta6 (numeclub CLUB.nume%TYPE)
AS
     TYPE tablou_indexat IS TABLE OF echipa%ROWTYPE INDEX BY PLS_INTEGER;
     echipe tablou_indexat;
     TYPE tablou_imbricat IS TABLE OF sportiv%ROWTYPE;
     sportivi tablou_imbricat:= tablou_imbricat();
     cnt NUMBER(4);
BEGIN
     SELECT e.id_echipa, e.id_departament, e.nume
     BULK COLLECT INTO  echipe
     FROM club c, departament d, echipa e
     WHERE e.id_departament=d.id_departament AND d.id_club = c.id_club AND
c.nume=numeclub;
     FOR i IN 1..echipe.COUNT LOOP
     DBMS_OUTPUT.put_line('-------------------------------------');
     DBMS_OUTPUT.put_line('Numele echipei: '||TO_CHAR(echipe(i).nume));
     cnt:=0;
        SELECT s.id_sportiv, s.id_echipa, s.nume, s.prenume, s.mail,
s.nr_telefon, s.nationalitate, s.data_nasterii
        BULK COLLECT INTO sportivi
        FROM sportiv s
        WHERE s.id_echipa = echipe(i).id_echipa;
        FOR j IN 1..sportivi.COUNT LOOP
           cnt:=cnt+1;
           DBMS_OUTPUT.put_line(TO_CHAR(cnt)||'.');
           DBMS_OUTPUT.put_line('Nume sportiv: '||sportivi(j).nume);
           DBMS_OUTPUT.put_line('Prenume sportiv: '||sportivi(j).prenume);
           DBMS_OUTPUT.put_line('Mail sportiv: '||sportivi(j).mail);
           DBMS_OUTPUT.put_line('Nr. telefon sportiv:
'||sportivi(j).nr_telefon);
           DBMS_OUTPUT.put_line('');
        END LOOP;
    END LOOP;
END;
/

BEGIN
cerinta6('CSM Targu Jiu');
END;
/
```



```
Numele echipei: Bright Reds
1.
Nume sportiv: Nitu
Prenume sportiv: Victor
Mail sportiv: vic.nitu@gmail.com
Nr. telefon sportiv: 0723433269

2.
Nume sportiv: Ioan
Prenume sportiv: Alex
Mail sportiv: alex.ioan@gmail.com
Nr. telefon sportiv: 0723435276

-------------------------------------
Numele echipei: Weekend Warriors
1.
Nume sportiv: Chirita
Prenume sportiv: Ruxandra
Mail sportiv: ruxy23@gmail.com
Nr. telefon sportiv: 0723432765

2.
Nume sportiv: Luita
Prenume sportiv: Gino
Mail sportiv: gino35@gmail.com
Nr. telefon sportiv: 0723432381

-------------------------------------
Numele echipei: The Elite Team
1.
Nume sportiv: Coman
Prenume sportiv: Laurentiu
Mail sportiv: laur.coman@gmail.com
Nr. telefon sportiv: 0765433265

2.
```



```
-------------------------------------
Numele echipei: The Elite Team
1.
Nume sportiv: Coman
Prenume sportiv: Laurentiu
Mail sportiv: laur.coman@gmail.com
Nr. telefon sportiv: 0765433265

2.
Nume sportiv: Mira
Prenume sportiv: Kiko
Mail sportiv: mira.kiko@gmail.com
Nr. telefon sportiv: 0764634261
```

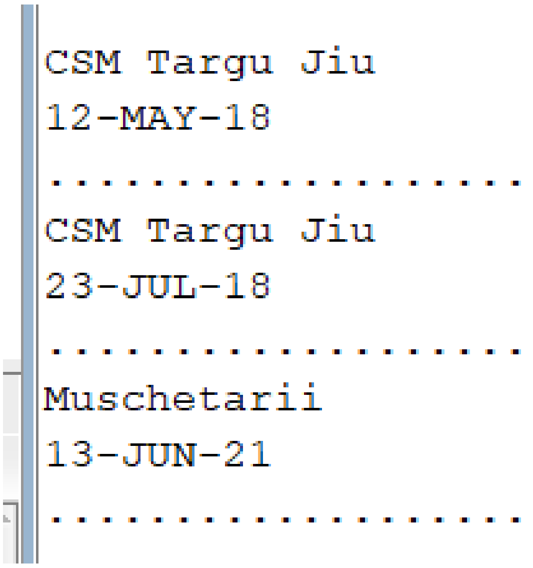
7. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent care să utilizeze 2 tipuri diferite de cursoare studiate, unul dintre acestea fiind cursor parametrizat. Apelați subprogramul.

Pentru un antenor dat(prin nume) afisati pentru competitiile la care a luat parte
impreuna cu echipa, clubul din care echipa cu un punctaj satisfacator face parte si
data desfasurarii competitiei. Daca exista mai multi antrenori cu acelasi nume se va alege primul.

```
CREATE OR REPLACE PROCEDURE cerinta7 (nume_antrenor antrenor.nume%TYPE)
AS
     id int;
     CURSOR compet( id antrenor.id_antrenor%TYPE) IS
                         SELECT c.id_echipa, c.data_desfasurarii
                        FROM competitie c, proba p
                        WHERE c.id_antrenor = id AND
c.puctaj_obtinut>=p.puctaj_minim AND c.id_proba=p.id_proba ;
    id_echipe competitie.id_echipa%TYPE;
    data competitie.data_desfasurarii%TYPE;
    is_found_rec BOOLEAN := FALSE;
BEGIN
    SELECT id_antrenor
   INTO id
   FROM antrenor
   WHERE nume=nume_antrenor AND ROWNUM=1;

   OPEN compet(id);
   LOOP
            FETCH compet INTO id_echipe, data;
            EXIT WHEN compet%NOTFOUND;
             is_found_rec := TRUE;
            FOR i IN (SELECT c.nume
                      FROM club c, departament d, echipa e
                      WHERE e.id_departament = d.id_departament AND
d.id_club=c.id_club AND e.id_echipa=id_echipe
                      ) LOOP
                       DBMS_OUTPUT.put_line(i.nume);
                       END LOOP;
            DBMS_OUTPUT.put_line(TO_CHAR(data));
             DBMS_OUTPUT.put_line('...................');
    END LOOP;
    CLOSE compet;
```

```
     IF NOT is_found_rec THEN
    DBMS_OUTPUT.put_line('Nu exista echipe');
 END IF;
END;
```

```
BEGIN

cerinta7('Luiza');

END;

/
```



```
CSM Targu Jiu
12-MAY-18
....................
CSM Targu Jiu
23-JUL-18
....................
Muschetarii
13-JUN-21
....................
```

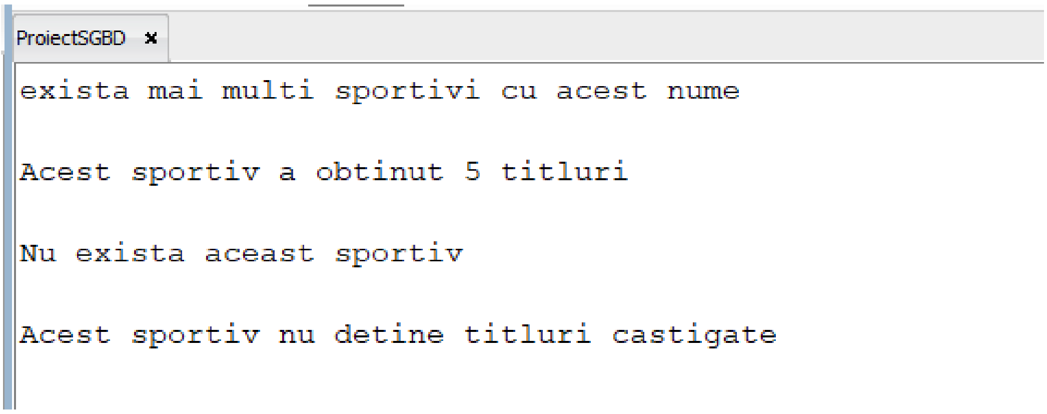
/

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

Pentru un sportiv dat dupa nume calculati numarul de titluri castigate

```
CREATE OR REPLACE FUNCTION cerinta8 (parametru sportiv.nume%TYPE) RETURN
NUMBER AS
       titluri NUMBER;
       TYPE tabel_id IS TABLE OF sportiv.id_sportiv%TYPE INDEX BY
PLS_INTEGER;
       id_uri tabel_id;
       Nu_exista_sportivi EXCEPTION;
       Nume_repetat EXCEPTION;
       nu_exista_titluri EXCEPTION;
BEGIN
      SELECT id_sportiv
      BULK COLLECT INTO id_uri
      FROM sportiv
      WHERE nume = parametru;
      IF id_uri.COUNT = 0 THEN
         RAISE Nu_exista_sportivi;
      END IF;
        IF id_uri.COUNT > 1 THEN
         RAISE Nume_repetat;
      END IF;
            SELECT  COUNT (s.id_sportiv)
            INTO titluri
            FROM sportiv s
            join sportiv_titlu st ON (s.id_sportiv = st.id_sportiv)
            join titlu t
            ON t.id_titlu = st.id_titlu
            WHERE s.id_sportiv = id_uri(1);
             IF titluri = 0 THEN
             RAISE nu_exista_titluri;
             ELSE
             RETURN titluri;
             END IF;
EXCEPTION
   WHEN nu_exista_sportivi THEN
      DBMS_OUTPUT.put_line('Nu exista aceast sportiv');
       RETURN -1;
    WHEN Nume_repetat THEN
      DBMS_OUTPUT.put_line('exista mai multi sportivi cu acest nume');
       RETURN -1;
    WHEN nu_exista_titluri THEN
      DBMS_OUTPUT.put_line('Acest sportiv nu detine titluri castigate');
       RETURN -1;
END;
/

 DECLARE
     nr NUMBER;
 BEGIN
     nr:=cerinta8('Ioan');
     IF(nr>=1) THEN
     DBMS_OUTPUT.put_line('Acest sportiv a obtinut '|| TO_CHAR(nr) || ' 
titluri');
     END IF;
 END;

 DECLARE
     nr NUMBER;
 BEGIN
     nr:=cerinta8('Ion');
     IF(nr>=1) THEN
     DBMS_OUTPUT.put_line('Acest sportiv a obtinut '|| TO_CHAR(nr) || ' 
titluri');
     END IF;
 END;

 DECLARE
     nr NUMBER;
 BEGIN
     nr:=cerinta8('Maria');
     IF(nr>=1) THEN
     DBMS_OUTPUT.put_line('Acest sportiv a obtinut '|| TO_CHAR(nr) || ' 
titluri');
```

```
     END IF;
 END;

 DECLARE
     nr NUMBER;
 BEGIN
     nr:=cerinta8('Luita');
     IF(nr>=1) THEN
     DBMS_OUTPUT.put_line('Acest sportiv a obtinut '|| TO_CHAR(nr) || '
titluri');
     END IF;
 END;
```

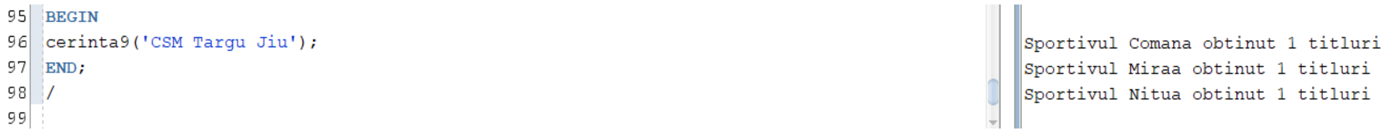


```
exista mai multi sportivi cu acest nume

Acest sportiv a obtinut 5 titluri

Nu exista aceast sportiv

Acest sportiv nu detine titluri castigate
```

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 comandăSQL5 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.

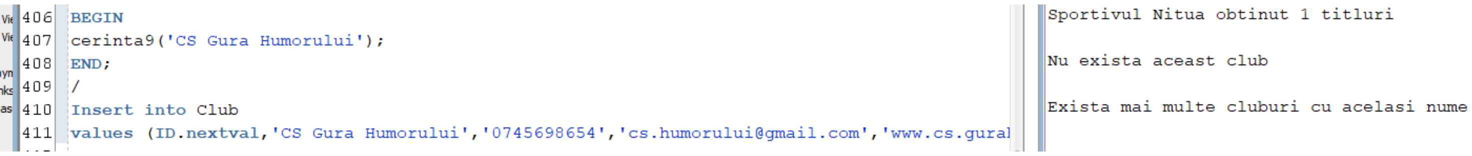
Pentru un club afisati numele sportivilor si nr de titluri dobandite

```
    CREATE OR REPLACE PROCEDURE cerinta9 (nume_club club.nume%TYPE) AS
    id_club_cautat int;
    TYPE tabel IS RECORD(
    id_sportiv_club sportiv.id_sportiv%TYPE,
      nr_titluri  NUMBER
      );
    TYPE sportiv_titlu IS TABLE OF tabel;
    sportiv_titluri sportiv_titlu;
    nume_sportiv VARCHAR(20);
    nu_exista EXCEPTION;
BEGIN
       SELECT id_club
       INTO id_club_cautat
       FROM club
       WHERE nume = nume_club;

        SELECT s.id_sportiv, COUNT(st.id_titlu)
        BULK COLLECT INTO sportiv_titluri
        FROM  sportiv s
        join sportiv_titlu st ON(st.id_sportiv= s.id_sportiv)
        join echipa e ON(s.id_echipa = e.id_echipa)
        join departament d ON (e.id_departament = d.id_departament)
        join club c ON (c.id_club = d.id_club)
        WHERE c.id_club = id_club_cautat
        GROUP BY s.id_sportiv;
        IF sportiv_titluri.COUNT=0 THEN
        RAISE nu_exista;
        END IF;
        FOR i IN 1..sportiv_titluri.COUNT LOOP
            SELECT nume
            INTO nume_sportiv
            FROM sportiv
            WHERE id_sportiv = sportiv_titluri(i).id_sportiv_club ;
            DBMS_OUTPUT.put_line('Sportivul '|| nume_sportiv || 'a obtinut
' || sportiv_titluri(i).nr_titluri || ' titluri'  );
        END LOOP;
EXCEPTION

   WHEN NO_DATA_FOUND THEN
      DBMS_OUTPUT.put_line('Nu exista aceast club');
   WHEN TOO_MANY_ROWS THEN
       DBMS_OUTPUT.put_line('Exista mai multe cluburi cu acelasi nume');
    WHEN nu_exista THEN
       DBMS_OUTPUT.put_line('Clubul nu are sportivi');
```



```
BEGIN
cerinta9('CSM Targu Jiu');
END;
/
```

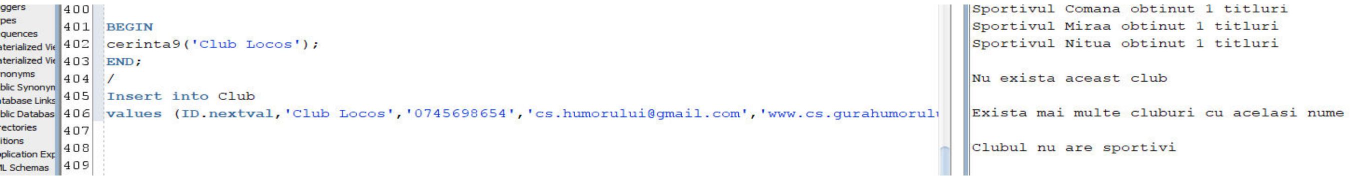
```
Sportivul Comana obtinut 1 titluri
Sportivul Miraa obtinut 1 titluri
Sportivul Nitua obtinut 1 titluri
```



```
BEGIN
cerinta9('CS Gura Humorului');
END;
/
Insert into Club
values (ID.nextval,'CS Gura Humorului','0745698654','cs.humorului@gmail.com','www.cs.gural
```

```
Sportivul Nitua obtinut 1 titluri

Nu exista aceast club

Exista mai multe cluburi cu acelasi nume
```

```
END;
```



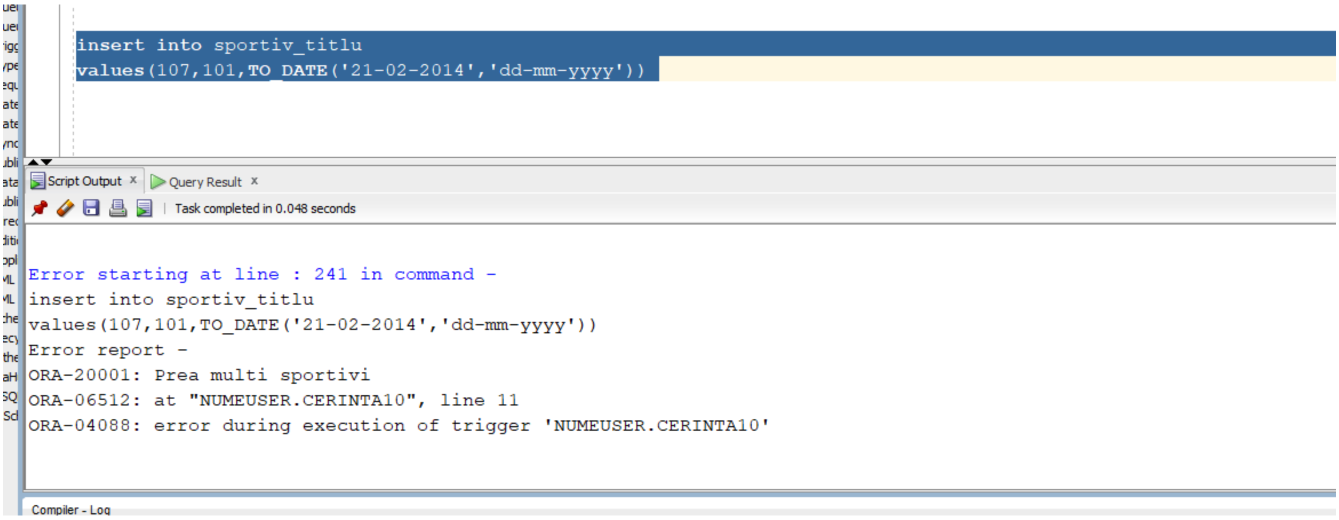
```
BEGIN
cerinta9('Club Locos');
END;
/
Insert into Club
values (ID.nextval,'Club Locos','0745698654','cs.humorului@gmail.com','www.cs.gurahumorulu
```

```
Sportivul Comana obtinut 1 titluri
Sportivul Miraa obtinut 1 titluri
Sportivul Nitua obtinut 1 titluri

Nu exista aceast club

Exista mai multe cluburi cu acelasi nume

Clubul nu are sportivi
```

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

IN tabela sportiv_titlu pot fi inserati maxim 6 sportivi diferiti

```
 CREATE OR REPLACE TRIGGER cerinta10
before INSERT ON sportiv_titlu
DECLARE
  sportivi NUMBER;
BEGIN
   SELECT  COUNT (DISTINCT (s.id_sportiv))
            INTO sportivi
            FROM sportiv s
            join sportiv_titlu st ON (s.id_sportiv = st.id_sportiv)
            join titlu t
            ON t.id_titlu = st.id_titlu;
            IF sportivi > 5 THEN
 RAISE_APPLICATION_ERROR(-20001,'Prea multi sportivi');
 END IF;
END;
```



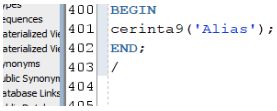
```
insert into sportiv_titlu
values(107,101,TO_DATE('21-02-2014','dd-mm-yyyy'))
```

```
Error starting at line : 241 in command -
insert into sportiv_titlu
values(107,101,TO_DATE('21-02-2014','dd-mm-yyyy'))
Error report -
ORA-20001: Prea multi sportivi
ORA-06512: at "NUMEUSER.CERINTA10", line 11
ORA-04088: error during execution of trigger 'NUMEUSER.CERINTA10'
```

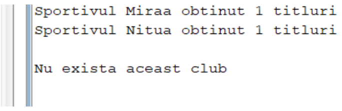
11. Definiți un trigger de tip LMD la nivel de linie. Declanșați trigger-ul.

Atunci cand un sportiv isi schimba echipa, suma acordata de sponsori va creste cu 100 pentru acea echipa

```
CREATE OR REPLACE TRIGGER cerinta11
        after UPDATE OF id_echipa ON sportiv
        FOR each ROW
DECLARE
```



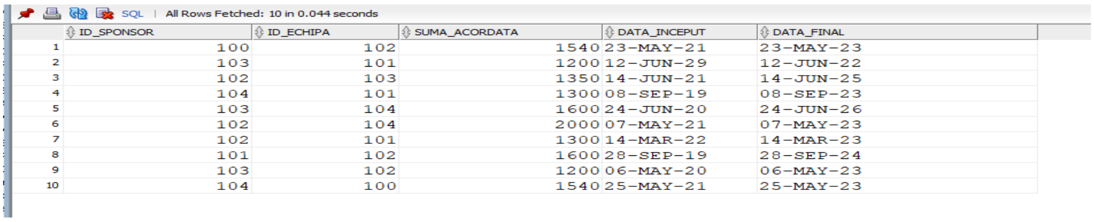
```
BEGIN
cerinta9('Alias');
END;
/
```



```
Sportivul Miraa obtinut 1 titluri
Sportivul Nitua obtinut 1 titluri

Nu exista aceast club
```

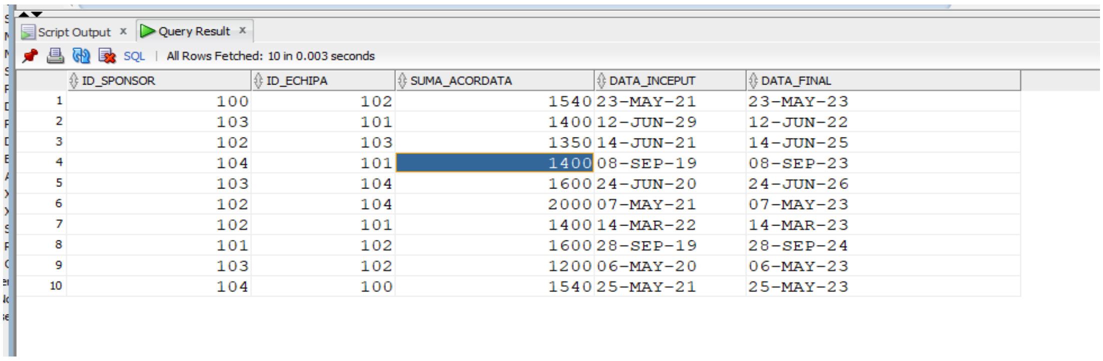
```
    exceptie EXCEPTION;
    TYPE suma IS TABLE OF echipa_sponsor.suma_acordata%TYPE INDEX BY
PLS_INTEGER;
    sume suma;
BEGIN
        IF(:old.id_echipa=:NEW.id_echipa)
            THEN RAISE exceptie;
            END IF;
        SELECT  suma_acordata
        BULK COLLECT INTO sume
        FROM echipa_sponsor
        WHERE id_echipa=(:NEW.id_echipa);
        FOR i IN 1..sume.COUNT LOOP
         UPDATE echipa_sponsor
         SET suma_acordata = sume(i)+100
         WHERE id_echipa = (:NEW.id_echipa);
         END LOOP;

EXCEPTION
        WHEN exceptie THEN
        RAISE_APPLICATION_ERROR (-20003, 'Sportivul nu s-a mutat din
echipa');
 END;


    UPDATE sportiv
    SET id_echipa=101
    WHERE id_sportiv=100;
```

Inainte



	ID_SPONSOR	ID_ECHIPA	SUMA_ACORDATA	DATA_INCEPUT	DATA_FINAL
1	100	102	1540	23-MAY-21	23-MAY-23
2	103	101	1200	12-JUN-29	12-JUN-22
3	102	103	1350	14-JUN-21	14-JUN-25
4	104	101	1300	08-SEP-19	08-SEP-23
5	103	104	1600	24-JUN-20	24-JUN-26
6	102	104	2000	07-MAY-21	07-MAY-23
7	102	101	1300	14-MAR-22	14-MAR-23
8	101	102	1600	28-SEP-19	28-SEP-24
9	103	102	1200	06-MAY-20	06-MAY-23
10	104	100	1540	25-MAY-21	25-MAY-23

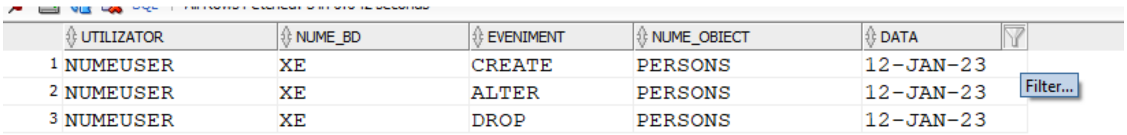
Dupa



	ID_SPONSOR	ID_ECHIPA	SUMA_ACORDATA	DATA_INCEPUT	DATA_FINAL
1	100	102	1540	23-MAY-21	23-MAY-23
2	103	101	1400	12-JUN-29	12-JUN-22
3	102	103	1350	14-JUN-21	14-JUN-25
4	104	101	1400	08-SEP-19	08-SEP-23
5	103	104	1600	24-JUN-20	24-JUN-26
6	102	104	2000	07-MAY-21	07-MAY-23
7	102	101	1400	14-MAR-22	14-MAR-23
8	101	102	1600	28-SEP-19	28-SEP-24
9	103	102	1200	06-MAY-20	06-MAY-23
10	104	100	1540	25-MAY-21	25-MAY-23

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

```
CREATE TABLE istoric
   (utilizator VARCHAR2(30),
    nume_bd     VARCHAR2(50),
    eveniment VARCHAR2(20),
    nume_obiect VARCHAR2(30),
    data   DATE);

CREATE OR REPLACE TRIGGER cerinta12
AFTER CREATE OR ALTER OR DROP ON SCHEMA
BEGIN
     INSERT INTO istoric
     VALUES (SYS.LOGIN_USER, SYS.DATABASE_NAME,
SYS.SYSEVENT,SYS.DICTIONARY_OBJ_NAME, SYSDATE);
END;


 CREATE TABLE Persons (
    PersonID int,
    LastName VARCHAR(255),
    FirstName VARCHAR(255),
    Address VARCHAR(255),
    City VARCHAR(255)
);

ALTER TABLE Persons DROP column City;

DROP TABLE Persons;

SELECT * FROM istoric;
```



	UTILIZATOR	NUME_BD	EVENIMENT	NUME_OBIECT	DATA
1	NUMEUSER	XE	CREATE	PERSONS	12-JAN-23
2	NUMEUSER	XE	ALTER	PERSONS	12-JAN-23
3	NUMEUSER	XE	DROP	PERSONS	12-JAN-23

13. Definiți un pachet care să conțină toate obiectele definite în cadrul proiectului.

```
CREATE OR REPLACE PACKAGE cerinta13 AS
     PROCEDURE cerinta6 (numeclub CLUB.nume%TYPE);
     PROCEDURE cerinta7 (nume_antrenor antrenor.nume%TYPE);
     FUNCTION cerinta8 (parametru sportiv.nume%TYPE) RETURN NUMBER;
     PROCEDURE cerinta9 (nume_club club.nume%TYPE);
END cerinta13;
/
CREATE OR REPLACE PACKAGE BODY cerinta13
AS
    PROCEDURE cerinta6 (numeclub CLUB.nume%TYPE)
AS
     TYPE tablou_indexat IS TABLE OF echipa%ROWTYPE INDEX BY PLS_INTEGER;
     echipe tablou_indexat;
     TYPE tablou_imbricat IS TABLE OF sportiv%ROWTYPE;
     sportivi tablou_imbricat:= tablou_imbricat();
     cnt NUMBER(4);
BEGIN
     SELECT e.id_echipa, e.id_departament, e.nume
     BULK COLLECT INTO  echipe
     FROM club c, departament d, echipa e
     WHERE e.id_departament=d.id_departament AND d.id_club = c.id_club AND
c.nume=numeclub;
     FOR i IN 1..echipe.COUNT LOOP
     DBMS_OUTPUT.put_line('--------------------------------------');
     DBMS_OUTPUT.put_line('Numele echipei: '||TO_CHAR(echipe(i).nume));
     cnt:=0;
        SELECT s.id_sportiv, s.id_echipa, s.nume, s.prenume, s.mail,
s.nr_telefon, s.nationalitate, s.data_nasterii
        BULK COLLECT INTO sportivi
        FROM sportiv s
        WHERE s.id_echipa = echipe(i).id_echipa;
        FOR j IN 1..sportivi.COUNT LOOP
           cnt:=cnt+1;
           DBMS_OUTPUT.put_line(TO_CHAR(cnt)||'.');
           DBMS_OUTPUT.put_line('Nume sportiv: '||sportivi(j).nume);
           DBMS_OUTPUT.put_line('Prenume sportiv: '||sportivi(j).prenume);
           DBMS_OUTPUT.put_line('Mail sportiv: '||sportivi(j).mail);
           DBMS_OUTPUT.put_line('Nr. telefon sportiv:
'||sportivi(j).nr_telefon);
           DBMS_OUTPUT.put_line('');
        END LOOP;
    END LOOP;
END cerinta6;
      PROCEDURE cerinta7 (nume_antrenor antrenor.nume%TYPE)
AS
     id int;
     CURSOR compet( id antrenor.id_antrenor%TYPE) IS
```

```
                         SELECT c.id_echipa, c.data_desfasurarii
                        FROM competitie c, proba p
                        WHERE c.id_antrenor = id AND
c.puctaj_obtinut>=p.puctaj_minim AND c.id_proba=p.id_proba ;
    id_echipe competitie.id_echipa%TYPE;
    data competitie.data_desfasurarii%TYPE;
    is_found_rec BOOLEAN := FALSE;
BEGIN
    SELECT id_antrenor
   INTO id
   FROM antrenor
   WHERE nume=nume_antrenor AND ROWNUM=1;

   OPEN compet(id);
   LOOP
            FETCH compet INTO id_echipe, data;
            EXIT WHEN compet%NOTFOUND;
             is_found_rec := TRUE;
            FOR i IN (SELECT c.nume
                      FROM club c, departament d, echipa e
                      WHERE e.id_departament = d.id_departament AND
d.id_club=c.id_club AND e.id_echipa=id_echipe
                      ) LOOP
                       DBMS_OUTPUT.put_line(i.nume);
                       END LOOP;
            DBMS_OUTPUT.put_line(TO_CHAR(data));
             DBMS_OUTPUT.put_line('...................');
    END LOOP;
    CLOSE compet;
     IF NOT is_found_rec THEN
    DBMS_OUTPUT.put_line('Nu exista echipe');
 END IF;
END cerinta7;
 FUNCTION cerinta8 (parametru sportiv.nume%TYPE) RETURN NUMBER AS
       titluri NUMBER;
       TYPE tabel_id IS TABLE OF sportiv.id_sportiv%TYPE INDEX BY
PLS_INTEGER;
       id_uri tabel_id;
       Nu_exista_sportivi EXCEPTION;
       Nume_repetat EXCEPTION;
       nu_exista_titluri EXCEPTION;
BEGIN
      SELECT id_sportiv
      BULK COLLECT INTO id_uri
      FROM sportiv
      WHERE nume = parametru;
      IF id_uri.COUNT = 0 THEN
         RAISE Nu_exista_sportivi;
      END IF;
        IF id_uri.COUNT > 1 THEN
```

```
         RAISE Nume_repetat;
      END IF;
            SELECT  COUNT (s.id_sportiv)
            INTO titluri
            FROM sportiv s
            join sportiv_titlu st ON (s.id_sportiv = st.id_sportiv)
            join titlu t
            ON t.id_titlu = st.id_titlu
            WHERE s.id_sportiv = id_uri(1);
             IF titluri = 0 THEN
             RAISE nu_exista_titluri;
             ELSE
             RETURN titluri;
             END IF;
EXCEPTION
   WHEN nu_exista_sportivi THEN
      DBMS_OUTPUT.put_line('Nu exista aceast sportiv');
       RETURN -1;
    WHEN Nume_repetat THEN
      DBMS_OUTPUT.put_line('exista mai multi sportivi cu acest nume');
       RETURN -1;
    WHEN nu_exista_titluri THEN
      DBMS_OUTPUT.put_line('Acest sportiv nu detine titluri castigate');
       RETURN -1;
END cerinta8;
   PROCEDURE cerinta9 (nume_club club.nume%TYPE) AS
    id_club_cautat int;
    TYPE tabel IS RECORD(
    id_sportiv_club sportiv.id_sportiv%TYPE,
      nr_titluri  NUMBER
      );
    TYPE sportiv_titlu IS TABLE OF tabel;
    sportiv_titluri sportiv_titlu;
    nume_sportiv VARCHAR(20);
    nu_exista EXCEPTION;
BEGIN
       SELECT id_club
       INTO id_club_cautat
       FROM club
       WHERE nume = nume_club;

        SELECT s.id_sportiv, COUNT(st.id_titlu)
        BULK COLLECT INTO sportiv_titluri
        FROM  sportiv s
        join sportiv_titlu st ON(st.id_sportiv= s.id_sportiv)
        join echipa e ON(s.id_echipa = e.id_echipa)
        join departament d ON (e.id_departament = d.id_departament)
        join club c ON (c.id_club = d.id_club)
        WHERE c.id_club = id_club_cautat
        GROUP BY s.id_sportiv;
        IF sportiv_titluri.COUNT=0 THEN
        RAISE nu_exista;
        END IF;
        FOR i IN 1..sportiv_titluri.COUNT LOOP
            SELECT nume
            INTO nume_sportiv
            FROM sportiv
            WHERE id_sportiv = sportiv_titluri(i).id_sportiv_club ;
            DBMS_OUTPUT.put_line('Sportivul '|| nume_sportiv || 'a obtinut
' || sportiv_titluri(i).nr_titluri || ' titluri'  );
        END LOOP;
EXCEPTION

   WHEN NO_DATA_FOUND THEN
      DBMS_OUTPUT.put_line('Nu exista aceast club');
   WHEN TOO_MANY_ROWS THEN
       DBMS_OUTPUT.put_line('Exista mai multe cluburi cu acelasi nume');
    WHEN nu_exista THEN
       DBMS_OUTPUT.put_line('Clubul nu are sportivi');
END cerinta9;
END cerinta13;
/

EXECUTE cerinta13.cerinta6('CSM Targu Jiu');
```

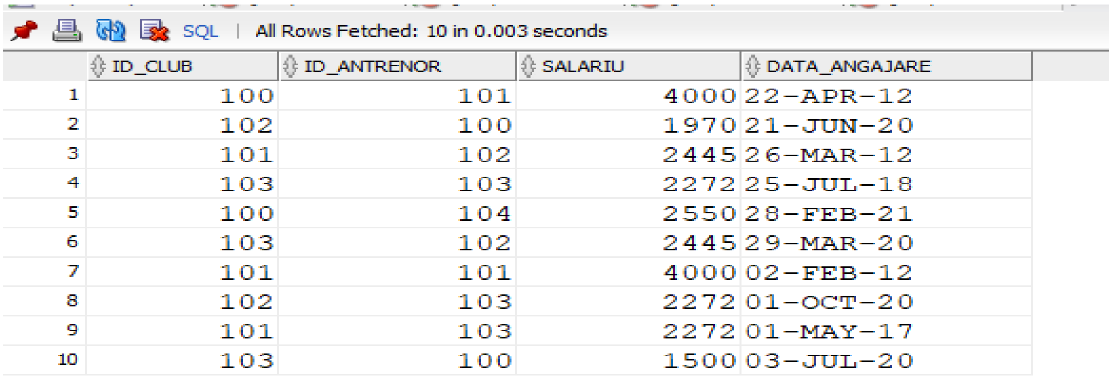
14. Definiți un pachet care să includă tipuri de date complexe și obiecte necesare unui flux de acțiuni integrate, specifice bazei de date definite (minim 2 tipuri de date, minim 2 funcții, minim 2 proceduri).

```
CREATE TYPE antrenori AS OBJECT (
    id int,
    nume_antrenor VARCHAR(20)
);
CREATE OR REPLACE TYPE antrenori_tip  AS TABLE OF antrenori ;

CREATE OR REPLACE PACKAGE cerinta14 AS

    FUNCTION AntrenoriPremii RETURN antrenori_tip;
    PROCEDURE GetInformatiiClub (numeclub CLUB.nume%TYPE);
    PROCEDURE UpdateSalariu ;
    FUNCTION Suma_Sponsori (numeclub CLUB.nume%TYPE) RETURN NUMBER;
 END cerinta14;
 /


CREATE OR REPLACE PACKAGE BODY cerinta14  AS
 FUNCTION AntrenoriPremii RETURN antrenori_tip AS
      antrenori_returnare  antrenori_tip := antrenori_tip();
       TYPE tabel IS RECORD(
    id_antrenor_club antrenor.id_antrenor%TYPE,
      nr_competitii  NUMBER
      );
      TYPE antrenor_cometitie IS TABLE OF tabel;
       antrenor_competitii antrenor_cometitie;
       no_data_found1 EXCEPTION;
       nume_antrenor VARCHAR(20);
  BEGIN
        SELECT id_antrenor, COUNT(id_antrenor)
        BULK COLLECT INTO antrenor_competitii
        FROM competitie
        GROUP BY id_antrenor;

        IF antrenor_competitii.COUNT = 0
        THEN RAISE no_data_found1;
        END IF;
        FOR i IN 1..antrenor_competitii.COUNT LOOP
            IF antrenor_competitii(i).nr_competitii>0 THEN
            SELECT nume
            INTO nume_antrenor
            FROM antrenor
            WHERE id_antrenor = antrenor_competitii(i).id_antrenor_club;
               antrenori_returnare.extend;
               antrenori_returnare(antrenori_returnare.COUNT) :=
antrenori(antrenor_competitii(i).id_antrenor_club,nume_antrenor);
            END IF;
            END LOOP;
            RETURN antrenori_returnare;
 EXCEPTION
   WHEN no_data_found1 THEN
      DBMS_OUTPUT.put_line('Nu exista antrenori');
  END AntrenoriPremii;
  PROCEDURE GetInformatiiClub (numeclub CLUB.nume%TYPE) AS
       id int;
     nr_locatii NUMBER;
     nr_antrenori NUMBER;
     nr_echipe NUMBER;
      TYPE tablou_indexat IS TABLE OF echipa%ROWTYPE INDEX BY PLS_INTEGER;
     echipe tablou_indexat;
     BEGIN
      SELECT id_club
     INTO id
     FROM club
     WHERE nume = numeclub;

   SELECT  COUNT ( c.id_club)
           INTO nr_locatii
            FROM club c
            join club_locatie cl ON (c.id_club = cl.id_club)
            WHERE c.id_club=id;
    SELECT  COUNT ( c.id_club)
```

```
           INTO nr_antrenori
            FROM club c
            join antrenor_club cl ON (c.id_club = cl.id_club)
            WHERE c.id_club=id;

    SELECT e.id_echipa, e.id_departament, e.nume
     BULK COLLECT INTO  echipe
     FROM club c, departament d, echipa e
     WHERE e.id_departament=d.id_departament AND d.id_club = c.id_club AND
c.nume=numeclub;

     nr_echipe := echipe.COUNT;

      DBMS_OUTPUT.put_line( 'Nr de locatii '|| TO_CHAR(nr_locatii));
      DBMS_OUTPUT.put_line( 'Nr de antrenori '|| TO_CHAR(nr_antrenori));
      DBMS_OUTPUT.put_line( 'Nr de echipe '|| TO_CHAR(nr_echipe));
 EXCEPTION
      WHEN NO_DATA_FOUND THEN
      DBMS_OUTPUT.put_line('Nu exista aceast club');
   WHEN TOO_MANY_ROWS THEN
       DBMS_OUTPUT.put_line('Exista mai multe cluburi cu acelasi nume');
END GetInformatiiClub;
  PROCEDURE UpdateSalariu  AS
   TYPE tablou_indexat IS TABLE OF NUMBER INDEX BY PLS_INTEGER;
   suma tablou_indexat;
   BEGIN
    FOR i IN ( SELECT id FROM TABLE(cerinta14.AntrenoriPremii)) LOOP
        SELECT salariu
        BULK COLLECT INTO suma
        FROM antrenor_club
        WHERE id_antrenor=i.id;
        FOR j IN 1..suma.COUNT LOOP
         FOR k IN (SELECT * FROM antrenor_club) LOOP
                        UPDATE antrenor_club
                         SET salariu = suma(j)+100
                         WHERE id_antrenor = i.id;
                         END LOOP;
         END LOOP;
    END LOOP;
EXCEPTION
      WHEN NO_DATA_FOUND THEN
      DBMS_OUTPUT.put_line('Nu exista antrenori');
END UpdateSalariu;
  FUNCTION Suma_Sponsori(numeclub CLUB.nume%TYPE) RETURN NUMBER AS
   TYPE tablou_indexat IS TABLE OF echipa%ROWTYPE INDEX BY PLS_INTEGER;
   TYPE tablou_indexat_numere IS TABLE OF NUMBER INDEX BY PLS_INTEGER;
   echipe tablou_indexat;
   suma_totala NUMBER;
   suma_curenta tablou_indexat_numere;
   exceptie EXCEPTION;
   aux NUMBER;
   BEGIN

    SELECT e.id_echipa, e.id_departament, e.nume
     BULK COLLECT INTO  echipe
     FROM club c, departament d, echipa e
     WHERE e.id_departament=d.id_departament AND d.id_club = c.id_club AND
c.nume=numeclub;

     IF echipe.COUNT=0 THEN
     RAISE NO_DATA_FOUND;
     END IF;
     suma_totala:=0;
     FOR i IN 1..echipe.COUNT LOOP
                SELECT suma_acordata
                BULK COLLECT INTO suma_curenta
                FROM echipa_sponsor
                WHERE id_echipa=echipe(i).id_echipa;
                FOR j IN 1..suma_curenta.COUNT LOOP
                aux:=suma_totala;
                suma_totala := suma_curenta(j)+aux;
                END LOOP;
            END LOOP;

        IF suma_totala=0 THEN
        RAISE exceptie;
        END IF;
       RETURN suma_totala;
EXCEPTION
      WHEN NO_DATA_FOUND THEN
      DBMS_OUTPUT.put_line('Nu exista antrenori');
       WHEN exceptie  THEN
      DBMS_OUTPUT.put_line('Clubul nu beneficiaza de sponsori');
END Suma_Sponsori;
END cerinta14;
/

EXECUTE cerinta14.getinformatiiclub('CSM Targu Jiu')
EXECUTE cerinta14.updatesalariu;

DECLARE

  a NUMBER;

BEGIN

    a:=cerinta14.suma_sponsori('CSM Targu Jiu');

    DBMS_OUTPUT.put_line('Suma totala a clubului ' || a);
```

```
END;
```

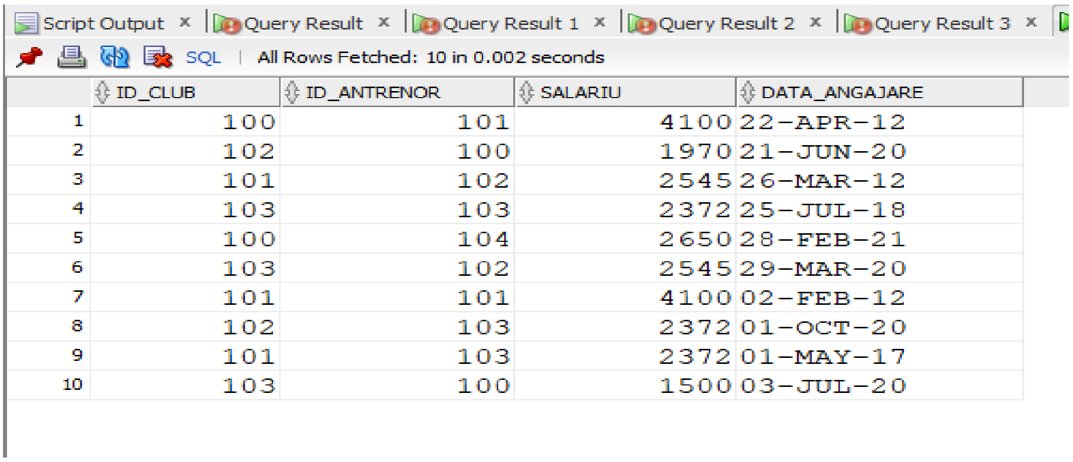
Inainte



All Rows Fetched: 10 in 0.003 seconds

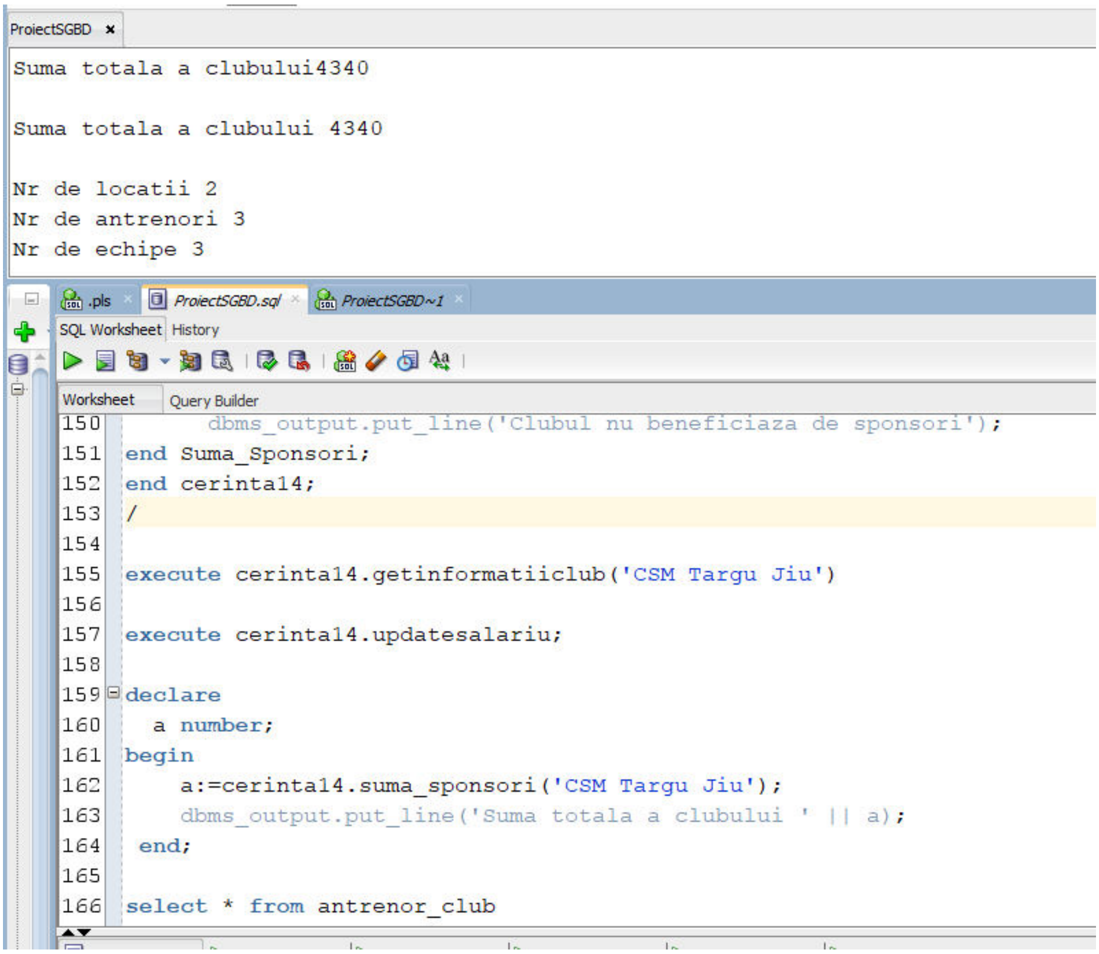
	ID_CLUB	ID_ANTRENOR	SALARIU	DATA_ANGAJARE
1	100	101	4000	22-APR-12
2	102	100	1970	21-JUN-20
3	101	102	2445	26-MAR-12
4	103	103	2272	25-JUL-18
5	100	104	2550	28-FEB-21
6	103	102	2445	29-MAR-20
7	101	101	4000	02-FEB-12
8	102	103	2272	01-OCT-20
9	101	103	2272	01-MAY-17
10	103	100	1500	03-JUL-20

Dupa



All Rows Fetched: 10 in 0.002 seconds

	ID_CLUB	ID_ANTRENOR	SALARIU	DATA_ANGAJARE
1	100	101	4100	22-APR-12
2	102	100	1970	21-JUN-20
3	101	102	2545	26-MAR-12
4	103	103	2372	25-JUL-18
5	100	104	2650	28-FEB-21
6	103	102	2545	29-MAR-20
7	101	101	4100	02-FEB-12
8	102	103	2372	01-OCT-20
9	101	103	2372	01-MAY-17
10	103	100	1500	03-JUL-20



```
Suma totala a clubului4340

Suma totala a clubului 4340

Nr de locatii 2
Nr de antrenori 3
Nr de echipe 3
```

```
      dbms_output.put_line('Clubul nu beneficiaza de sponsori');
end Suma_Sponsori;
end cerinta14;
/

execute cerinta14.getinformatiiclub('CSM Targu Jiu')

execute cerinta14.updatesalariu;

declare
  a number;
begin
    a:=cerinta14.suma_sponsori('CSM Targu Jiu');
    dbms_output.put_line('Suma totala a clubului ' || a);
 end;

select * from antrenor_club
```