

SISTEM DE GESTIUNE
- ECHIPA DE FOTBAL -
PROIECT - SGBD

Toma David

- grupa 231

An 2023-2024, semestrul I

Cuprins:

1. Prezențați pe scurt baza de date (utilitatea ei).
2. Realizați diagrama entitate-relație (ERD): entitățile, relațiile și atributele trebuie definite în limba

română (vezi curs SGBD / model de diagrama ERD; nu se va accepta alt format).

3. Pornind de la diagrama entitate-relație realizați diagrama conceptuală a modelului propus, integrând

toate atributele necesare: entitățile, relațiile și atributele trebuie definite în limba română.

4. Implementați în Oracle diagrama conceptuală realizată: definiți toate tabelele, definind toate constrângerile de integritate necesare (chei primare, cheile externe etc).

5. Adăugați informații coerente în tabelele create (minim 5 înregistrări pentru fiecare entitate independentă; minim 10 înregistrări pentru tabela asociativă).

6. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un subprogram stocat independent care să utilizeze toate cele 3 tipuri de colecții studiate. Apelați subprogramul.

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, dependent de celălalt cursor. Apelați subprogramul.

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 SQL 3 dintre tabelele definite. Definiți minim 2 excepții proprii. Apelați subprogramul astfel încât să evidențiați toate cazurile definite și tratate.

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

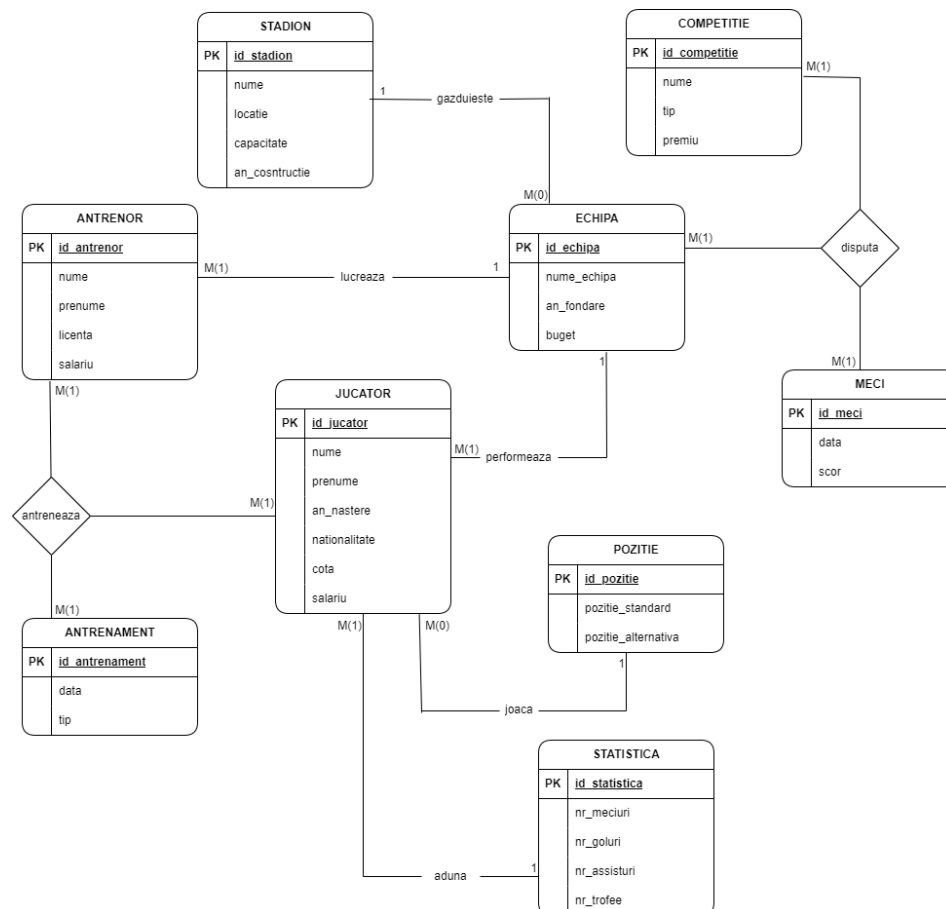
10. Definiți un trigger de tip LMD la nivel de comandă. Declanșați trigger-ul.
11. Definiți un trigger de tip LMD la nivel de linie. Declanșați trigger-ul.
12. Definiți un trigger de tip LDD. Declanșați trigger-ul.

1 – PREZENTARE PE SCURT A BAZE DE DATE

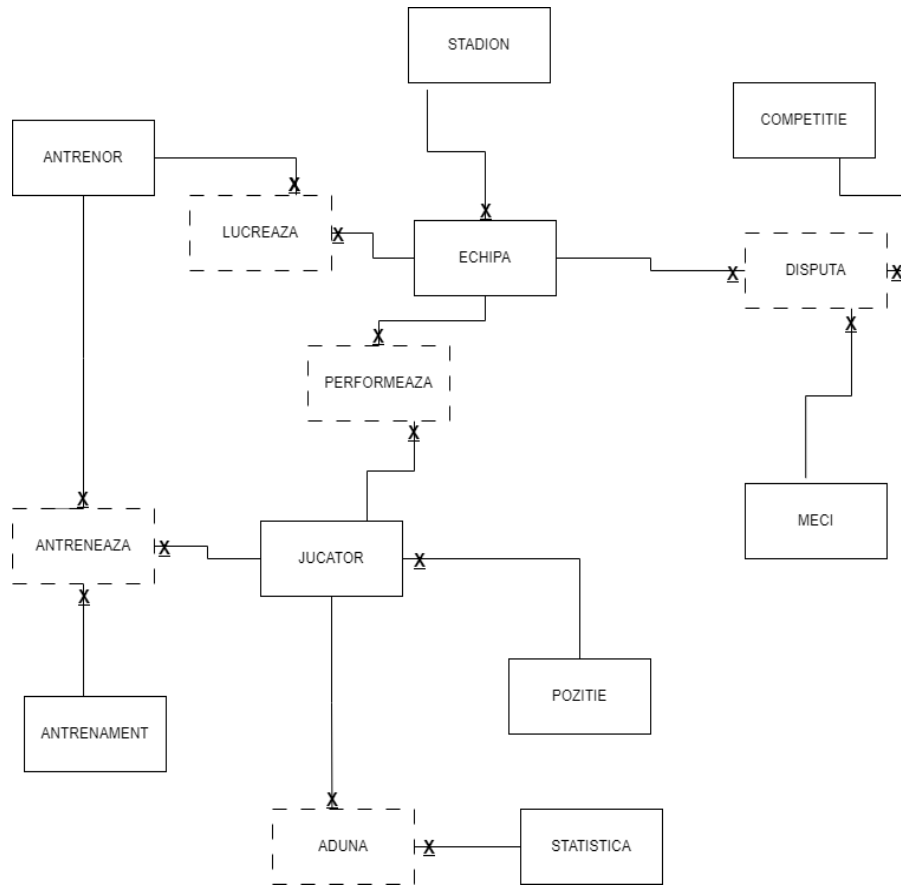
În cadrul acestui proiect, tema aleasă a fost “Gestiunea unei echipe de fotbal”. Scopul acestui proiect este de a ușura și a eficientiza administrarea unei echipe de fotbal și a resurselor aferente acesteia.

- Echipa își joacă meciurile pe un stadion
- Un stadion poate gazdui mai multe echipe
- Jucătorii performează la o echipă
- Antrenorii lucrează la o echipă
- Antrenorii antrenează jucătorii
- O echipă poate avea mai mulți jucători și mai mulți antrenori
- Un antrenament poate alinia mai mulți jucători și mai mulți antrenori
- Un jucător evoluează pe o poziție
- Un jucător dispune de o statistică
- O echipă dispută meciuri
- Meciurile se desfășoară în cadrul unei competiții, din care, firește, echipa face parte

2 – DIAGRAMA ENTITATE - RELATIE



3 – DIAGRAMA CONCEPTUALA



4 – CREAREA TABELELOR + CONSTRANGERILOR

-- stadion

create table stadion (

id_stadion number(5) constraint pk_stadion primary key,

nume varchar2(20) unique,

locatie varchar2(20),

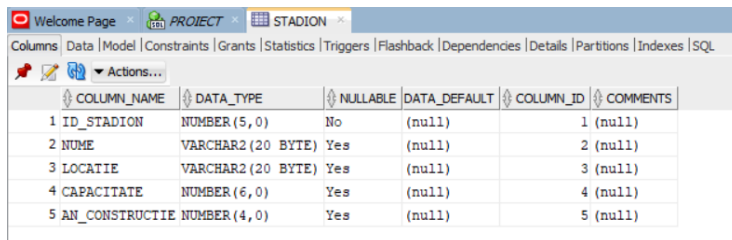
capacitate number(6) constraint check_capacitate check(capacitate >= 0),

an_constructie number(4)

);

```
-- stadion

create table stadion (
    id_stadion number(5) constraint pk_stadion primary key,
    nume varchar2(20) unique,
    locatie varchar2(20),
    capacitate number(6) constraint check_capacitate check(capacitate >= 0),
    an_constructie number(4)
);
```



The screenshot shows the Oracle SQL Developer interface with the 'STADION' table selected. The 'Columns' tab is active, displaying the following table structure:

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_STADION	NUMBER(5,0)	No	(null)	1 (null)	
2 NUME	VARCHAR2(20 BYTE)	Yes	(null)	2 (null)	
3 LOCATIE	VARCHAR2(20 BYTE)	Yes	(null)	3 (null)	
4 CAPACITATE	NUMBER(6,0)	Yes	(null)	4 (null)	
5 AN_CONSTRUCTIE	NUMBER(4,0)	Yes	(null)	5 (null)	

-- echipa

create table echipa(

id echipa number(5) constraint pk_echipa primary key,

id_stadion number(5) constraint fk_echipa_stadion references stadion(id_stadion),

nume_echipa varchar(20) unique,

an_fondare number(4),

buget number(10)

);

```
-- echipa

create table echipa(
    id_echipa number(5) constraint pk_echipa primary key,
    id_stadion number(5) constraint fk_echipa_stadion references stadion(id_stadion),
    nume_echipa varchar(20) unique,
    an_fondare number(4),
    buget number(10)
);
```

Welcome Page

PROJECT

ECHIPA

Columns

Data

Model

Constraints

Grants

Statistics

Triggers

Flashback

Dependencies

Details

Partitions

Indexes

SQL

Actions...

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_ECHIPA	NUMBER(5,0)	No	(null)	1 (null)	
2 ID_STADION	NUMBER(5,0)	Yes	(null)	2 (null)	
3 NUME_ECHIPA	VARCHAR2(20 BYTE)	Yes	(null)	3 (null)	
4 AN_FONDARE	NUMBER(4,0)	Yes	(null)	4 (null)	
5 BUGET	NUMBER(10,0)	Yes	(null)	5 (null)	

-- antrenor

create table antrenor(

id_antrenor number(5) constraint pk_antrenor primary key,

nume varchar2(20),

prenume varchar2(20),

licenta char(2) constraint chk_licenta_valori check (licenta in ('DA', 'NU'))

);

```
-- antrenor
create table antrenor(
  id_antrenor number(5) constraint pk_antrenor primary key,
  nume varchar2(20),
  prenume varchar2(20),
  licenta char(2) constraint chk_licenta_valori check (licenta in ('DA', 'NU'))
);
```

Welcome Page		PROJECT		ANTRENOR	
Columns	Data	Model	Constraints	Grants	Statistics

-- antrenament

create table antrenament(

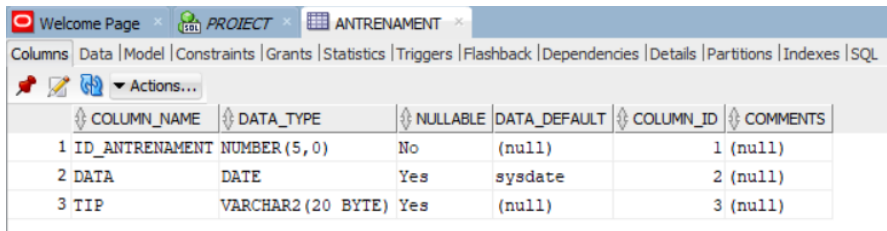
id_antrenament number(5) constraint pk_antrenament primary key,

data date default sysdate,

tip varchar2(20)

```
-- antrenament
create table antrenament(
  id_antrenament number(5) constraint pk_antrenament primary key,
  data date default sysdate,
  tip varchar2(20)
);
```

);



Columns | Data | Model | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_ANTRENAMENT	NUMBER(5,0)	No	(null)	1	(null)
2	DATA	DATE	Yes	sysdate	2	(null)
3	TIP	VARCHAR2(20 BYTE)	Yes	(null)	3	(null)

-- pozitie

```
-- pozitie
create table pozitie(
  id_pozitie number(5) constraint pk_pozitie primary key,
  pozitie_standard varchar(2) constraint check_pozitie_1 check(pozitie_standard in ('GK','LB','CB','RB','CDM','CM','CAM','LM','RM','LW','RW','CF','ST')),
  pozitie_alternativa varchar(2) constraint check_pozitie_2 check(pozitie_alternativa in ('GK','LB','CB','RB','CDM','CM','CAM','LM','RM','LW','RW','CF','ST'))
);
```

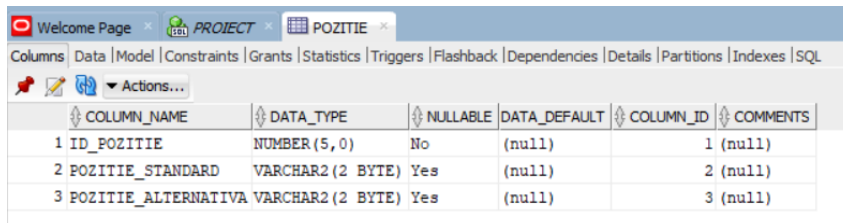
create table pozitie(

id_pozitie number(5) constraint pk_pozitie primary key,

pozitie_standard varchar(2) constraint check_pozitie_1 check(pozitie_standard in ('GK','LB','CB','RB','CDM','CM','CAM','LM','RM','LW','RW','CF','ST')),

pozitie_alternativa varchar(2) constraint check_pozitie_2 check(pozitie_alternativa in ('GK','LB','CB','RB','CDM','CM','CAM','LM','RM','LW','RW','CF','ST'))

);



Columns | Data | Model | Constraints | Grants | Statistics | Triggers | Flashback | Dependencies | Details | Partitions | Indexes | SQL

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_POZITIE	NUMBER(5,0)	No	(null)	1	(null)
2	POZITIE_STANDARD	VARCHAR2(2 BYTE)	Yes	(null)	2	(null)
3	POZITIE_ALTERNATIVA	VARCHAR2(2 BYTE)	Yes	(null)	3	(null)

-- jucator

```
-- jucator
create table jucator(
  id_jucator number(5) constraint pk_jucator primary key,
  id_pozitie number(5) constraint fk_jucator_pozitie references pozitie(id_pozitie),
  nume varchar2(20),
  prenume varchar2(20),
  an_nastere number(4) constraint check_an check(an_nastere >= 2023),
  nationalitate varchar2(20),
  cota number(10) constraint check_cota check(cota >= 0),
  salariu number(10)
);
```

create table jucator(

id_jucator number(5) constraint pk_jucator primary key,

id_pozitie number(5) constraint fk_jucator__pozitie references pozitie(id_pozitie),

```

nume varchar2(20),
prenume varchar2(20),
an_nastere number(4) constraint check_an check(an_nastere <= 2023),
nationalitate varchar2(20),
cota number(10) constraint check_cota check(cota >= 0),
salariu number(10)
);

```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_JUCATOR	NUMBER(5,0)	No	(null)	1 (null)	
2 ID_POZITIE	NUMBER(5,0)	Yes	(null)	2 (null)	
3 NUME	VARCHAR2(20 BYTE)	Yes	(null)	3 (null)	
4 PRENUME	VARCHAR2(20 BYTE)	Yes	(null)	4 (null)	
5 AN_NASTERE	NUMBER(4,0)	Yes	(null)	5 (null)	
6 NATIONALITATE	VARCHAR2(20 BYTE)	Yes	(null)	6 (null)	
7 COTA	NUMBER(10,0)	Yes	(null)	7 (null)	
8 SALARIU	NUMBER(10,0)	Yes	(null)	8 (null)	

-- statistica

create table statistica(

```

-- statistica
create table statistica(
  id_statistica number(5) constraint pk_statistica primary key,
  nr_meciuri number(4) constraint check_nr_meciuri check(nr_meciuri >= 0),
  nr_goluri number(4) constraint check_nr_goluri check(nr_goluri >= 0),
  nr_assisturi number(4) constraint check_nr_assisturi check(nr_assisturi >= 0),
  nr_trofee number(2) constraint check_nr_trofee check(nr_trofee >= 0)
);

```

```

id_statistica number(5) constraint pk_statistica primary key,
nr_meciuri number(4) constraint check_nr_meciuri check(nr_meciuri >= 0),
nr_goluri number(4) constraint check_nr_goluri check(nr_goluri >= 0),
nr_assisturi number(4) constraint check_nr_assisturi check(nr_assisturi >= 0),
nr_trofee number(2) constraint check_nr_trofee check(nr_trofee >= 0)
);

```

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_STATISTICA	NUMBER(5,0)	No	(null)	1 (null)	
2 NR_MEIURI	NUMBER(4,0)	Yes	(null)	2 (null)	
3 NR_GOLURI	NUMBER(4,0)	Yes	(null)	3 (null)	
4 NR_ASSISTURI	NUMBER(4,0)	Yes	(null)	4 (null)	
5 NR_TROFEE	NUMBER(2,0)	Yes	(null)	5 (null)	

-- competitie

create table competitie(

id_competitie number(5) constraint pk_competitie primary key,

nume varchar2(30),

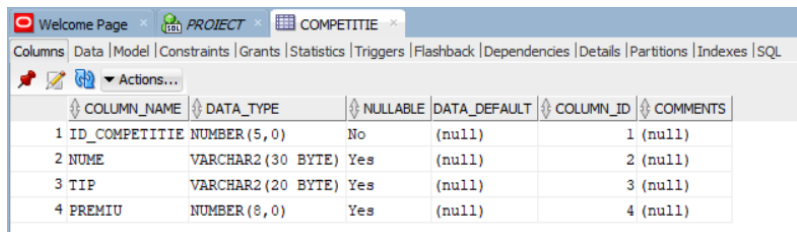
tip varchar2(20),

premiu number(8)

);

```
-- competitie

create table competitie(
  id_competitie number(5) constraint pk_competitie primary key,
  nume varchar2(30),
  tip varchar2(20),
  premiu number(8)
);
```



The screenshot shows the 'COMPETITIE' table structure in SQL Developer. The table has four columns: ID_COMPETITIE (NUMBER(5,0)), NUME (VARCHAR2(30 BYTE)), TIP (VARCHAR2(20 BYTE)), and PREMIU (NUMBER(8,0)). All columns are nullable and have a default value of null. The primary key is on ID_COMPETITIE.

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_COMPETITIE	NUMBER(5,0)	No	(null)	1 (null)	
2 NUME	VARCHAR2(30 BYTE)	Yes	(null)	2 (null)	
3 TIP	VARCHAR2(20 BYTE)	Yes	(null)	3 (null)	
4 PREMIU	NUMBER(8,0)	Yes	(null)	4 (null)	

-- meci

create table meci(

id_meci number(5) constraint pk_meci primary key,

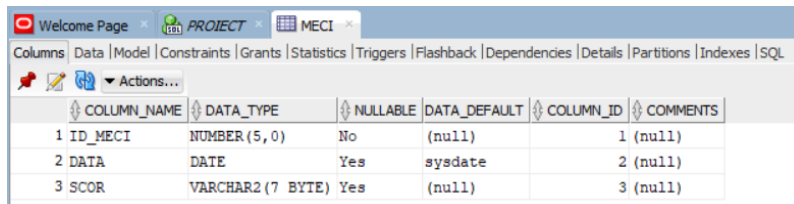
data date default sysdate,

scor varchar(7) constraint check_scor check(scor like '% - %')

);

```
-- meci

create table meci(
  id_meci number(5) constraint pk_meci primary key,
  data date default sysdate,
  scor varchar(7) constraint check_scor check(scor like '% - %')
);
```



The screenshot shows the 'MECI' table structure in SQL Developer. The table has three columns: ID_MECI (NUMBER(5,0)), DATA (DATE), and SCOR (VARCHAR2(7 BYTE)). ID_MECI is the primary key. DATA has a default value of sysdate. SCOR has a check constraint to ensure it follows the format '% - %'.

COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_MECI	NUMBER(5,0)	No	(null)	1 (null)	
2 DATA	DATE	Yes	sysdate	2 (null)	
3 SCOR	VARCHAR2(7 BYTE)	Yes	(null)	3 (null)	

-- Tabelul asociativ PERFORMEAZA

create table performeaza(

id echipa number(5),

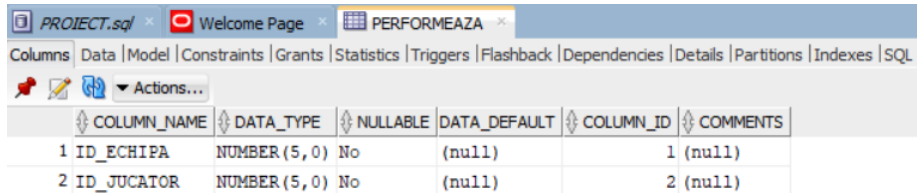
```
-- Tabelul asociativ PERFORMEAZA

create table performeaza(
  id echipa number(5),
  id jucator number(5),
  constraint pk_performeaza primary key (id echipa, id jucator),
  constraint fk_performeaza echipa foreign key (id echipa) references echipa(id echipa),
  constraint fk_performeaza jucator foreign key (id jucator) references jucator(id jucator)
);
```

```

id_jucator number(5),
constraint pk_performeaza primary key (id echipa, id_jucator),
constraint fk_performeaza echipa foreign key (id echipa) references echipa(id echipa),
constraint fk_performeaza jucator foreign key (id_jucator) references jucator(id_jucator)
);

```



The screenshot shows a database management interface with tabs for PROJECT.sql, Welcome Page, and PERFORMEAZA. The 'Columns' tab is active, displaying a table with 6 columns: COLUMN_NAME, DATA_TYPE, NULLABLE, DATA_DEFAULT, COLUMN_ID, and COMMENTS. The table contains two rows of data.

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_ECHIPA	NUMBER(5,0)	No	(null)	1	(null)
2	ID_JUCATOR	NUMBER(5,0)	No	(null)	2	(null)

-- Tabelul asociativ **LUCREAZA**

```
create table lucreaza(
```

```
    id echipa number(5),
```

```
    id antrenor number(5),
```

```
    constraint pk_lucreaza primary key (id echipa, id antrenor),
```

```
    constraint fk_lucreaza echipa foreign key (id echipa) references echipa(id echipa),
```

```
    constraint fk_lucreaza antrenor foreign key (id antrenor) references antrenor(id antrenor)
```

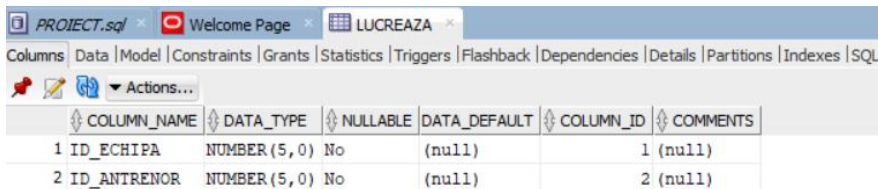
```
);
```

```
-- Tabelul asociativ LUCREAZA
```

```

create table lucreaza(
    id echipa number(5),
    id antrenor number(5),
    constraint pk_lucreaza primary key (id echipa, id antrenor),
    constraint fk_lucreaza echipa foreign key (id echipa) references echipa(id echipa),
    constraint fk_lucreaza antrenor foreign key (id antrenor) references antrenor(id antrenor)
);

```



The screenshot shows a database management interface with tabs for PROJECT.sql, Welcome Page, and LUCREAZA. The 'Columns' tab is active, displaying a table with 6 columns: COLUMN_NAME, DATA_TYPE, NULLABLE, DATA_DEFAULT, COLUMN_ID, and COMMENTS. The table contains two rows of data.

	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_ECHIPA	NUMBER(5,0)	No	(null)	1	(null)
2	ID_ANTRENOR	NUMBER(5,0)	No	(null)	2	(null)

-- Tabelul asociativ **ADUNA**

```

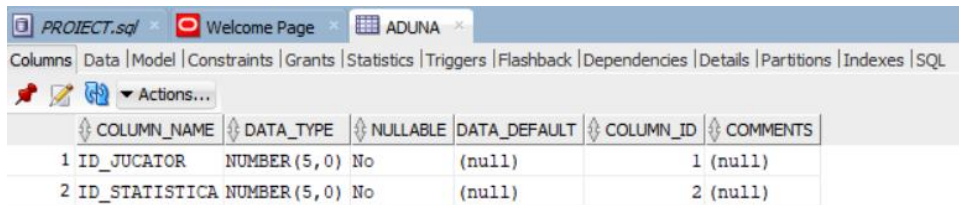
create table aduna(
    id_jucator number(5),
    id_statistica number(5),
    constraint pk_aduna primary key (id_jucator, id_statistica),
    constraint fk_aduna_jucator foreign key (id_jucator) references jucator(id_jucator),
    constraint fk_aduna_statistica foreign key (id_statistica) references statistica(id_statistica)
);

```

```

create table aduna(
    id_jucator number(5),
    id_statistica number(5),
    constraint pk_aduna primary key (id_jucator, id_statistica),
    constraint fk_aduna_jucator foreign key (id_jucator) references jucator(id_jucator),
    constraint fk_aduna_statistica foreign key (id_statistica) references statistica(id_statistica)
);

```



	COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1	ID_JUCATOR	NUMBER(5,0)	No	(null)	1 (null)	
2	ID_STATISTICA	NUMBER(5,0)	No	(null)	2 (null)	

-- Tabelul asociativ **ANTRENEAZA**

```
create table antreneaza(
```

```
    id_jucator number(5),
```

```
    id_antrenor number(5),
```

```
    id_antrenament number(5),
```

```
    constraint pk_antreneaza primary key (id_jucator, id_antrenor, id_antrenament),
```

```
    constraint fk_antreneaza_jucator foreign key (id_jucator) references jucator(id_jucator),
```

```
    constraint fk_antreneaza_antrenor foreign key (id_antrenor) references antrenor(id_antrenor),
```

```
    constraint fk_antreneaza_antrenament foreign key (id_antrenament) references antrenament(id_antrenament)
```

```
);
```

```
-- Tabelul asociativ ANTRENEAZA
```

```

create table antreneaza(
    id_jucator number(5),
    id_antrenor number(5),
    id_antrenament number(5),
    constraint pk_antreneaza primary key (id_jucator, id_antrenor, id_antrenament),
    constraint fk_antreneaza_jucator foreign key (id_jucator) references jucator(id_jucator),
    constraint fk_antreneaza_antrenor foreign key (id_antrenor) references antrenor(id_antrenor),
    constraint fk_antreneaza_antrenament foreign key (id_antrenament) references antrenament(id_antrenament)
);

```

PROJECT.sql					
Columns					
COLUMN_NAME	DATA...	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_JUCATOR	NUMBER(5,0)	No	(null)	1 (null)	
2 ID_ANTRENOR	NUMBER(5,0)	No	(null)	2 (null)	
3 ID_ANTRENAMENT	NUMBER(5,0)	No	(null)	3 (null)	

-- Tabelul asociativ **DISPUTA**

create table disputa(

id echipa number(5),

id competitie number(5),

id meci number(5),

constraint pk_disputa primary key (id echipa, id competitie, id meci),

constraint fk_disputa_echipa foreign key (id echipa) references echipa(id echipa),

constraint fk_disputa_competitie foreign key (id competitie) references competitie(id competitie),

constraint fk_disputa_meci foreign key (id meci) references meci(id meci)

);

```
-- Tabelul asociativ DISPUTA

create table disputa(
    id echipa number(5),
    id competitie number(5),
    id meci number(5),
    constraint pk_disputa primary key (id echipa, id competitie, id meci),
    constraint fk_disputa_echipa foreign key (id echipa) references echipa(id echipa),
    constraint fk_disputa_competitie foreign key (id competitie) references competitie(id competitie),
    constraint fk_disputa_meci foreign key (id meci) references meci(id meci)
);
```

PROJECT.sql					
Columns					
COLUMN_NAME	DATA_TYPE	NULLABLE	DATA_DEFAULT	COLUMN_ID	COMMENTS
1 ID_ECHIPA	NUMBER(5,0)	No	(null)	1 (null)	
2 ID_COMPETITIE	NUMBER(5,0)	No	(null)	2 (null)	
3 ID_MECI	NUMBER(5,0)	No	(null)	3 (null)	

5 - INSERAREA DATELOR COERENTE ÎN TABELE

--- secventa + inserari -> STADION

create sequence id_stadion start with 1 increment by 1 minvalue 0 maxvalue 9999 nocycle;

insert into stadion values(id_stadion.nextval, 'Camp Nou', 'Barcelona', 99000, 1957)

insert into stadion values(id_stadion.nextval, 'Ghencea', 'Bucharest', 30000, 2020);

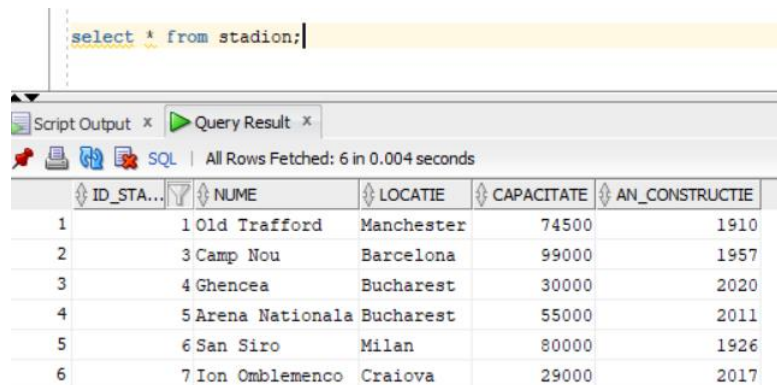
insert into stadion values(id_stadion.nextval, 'Arena Nationala', 'Bucharest', 55000, 2011);

insert into stadion values(id_stadion.nextval, 'San Siro', 'Milan', 80000, 1926);

insert into stadion values(id_stadion.nextval, 'Ion Omblemenco', 'Craiova', 29000, 2017);

insert into stadion values(id_stadion.nextval, 'Old Trafford', 'Manchester', 74500, 1910);

```
select * from stadion;
```



The screenshot shows a database query result for the 'stadion' table. The query is 'select * from stadion;'. The result is displayed in a table with 6 rows and 5 columns: ID_STA..., NUME, LOCATIE, CAPACITATE, and AN_CONSTRUCTIE. The data is as follows:

ID_STA...	NUME	LOCATIE	CAPACITATE	AN_CONSTRUCTIE
1	1 Old Trafford	Manchester	74500	1910
2	3 Camp Nou	Barcelona	99000	1957
3	4 Ghencea	Bucharest	30000	2020
4	5 Arena Nationala	Bucharest	55000	2011
5	6 San Siro	Milan	80000	1926
6	7 Ion Omblemenco	Craiova	29000	2017

--- secventa + inserari -> ECHIPA

create sequence id_echipa start with 1 increment by 1 minvalue 0 maxvalue 9999 nocycle;

insert into echipa values(id_echipa.nextval,1,'Manchester United',1903,80000000);

insert into echipa values(id_echipa.nextval, 7, 'CSU Craiova', 2013, 4000000);

insert into echipa values(id_echipa.nextval, 3, 'FC Barcelona', 1899, 95000000);

insert into echipa values(id_echipa.nextval, 5, 'FC Steaua Bucuresti', 1947, 15000000);

insert into echipa values(id_echipa.nextval, 7, 'U Craiova 1948', 1948, 2000000);

insert into echipa values(id_echipa.nextval, 4, 'CSA Steaua', 2017, 9500000);

insert into echipa values(id_echipa.nextval, 6, 'AC Milan', 1899, 90000000);

insert into echipa values(id echipa.nextval, 6, 'Inter Milano', 1908, 85000000);

select * from echipa;

ID_ECHIPA	ID_STADION	NUME_EC...	AN_FOND...	BUGET
1	1	1 Manches...	1903	80000000
2	2	7 CSU Cra...	2013	40000000
3	7	3 FC Barc...	1899	95000000
4	9	5 FC Stea...	1947	15000000
5	10	7 U Craio...	1948	2000000
6	12	4 CSA Steaua	2017	9500000
7	13	6 AC Milan	1899	90000000
8	14	6 Inter M...	1908	85000000

--- secventa + inserari → ANTRENOR

create sequence id_antrenor start with 1 increment by 1 minvalue 0 maxvalue 9999 nocycle;

insert into antrenor values(id_antrenor.nextval, 'Ranar', 'Alberto', 'NU');

insert into antrenor values(id_antrenor.nextval, 'Sumudica', 'Marius', 'DA');

insert into antrenor values(id_antrenor.nextval, 'Smith', 'John', 'DA');

insert into antrenor values(id_antrenor.nextval, 'Garcia', 'Martin', 'NU');

insert into antrenor values(id_antrenor.nextval, 'Brown', 'Michael', 'DA');

insert into antrenor values(id_antrenor.nextval, 'Lopez', 'Alexander', 'NU');

insert into antrenor values(id_antrenor.nextval, 'Lopez', 'Adrian Munteanu', 'DA');

insert into antrenor values(id_antrenor.nextval, 'Nita', 'Robert', 'NU');

insert into antrenor values(id_antrenor.nextval, 'Stoica', 'Mihai', 'NU');

insert into antrenor values(id_antrenor.nextval, 'Lopez', 'Francisc', 'DA');

select * from antrenor;

ID_ANTRENOR	NUME	PRENUME	LICENTA
1	1 Ranar	Alberto	NU
2	2 Sumudica	Marius	DA
3	3 Smith	John	DA
4	9 Brown	Michael	DA
5	10 Lopez	Alexander	NU
6	11 Garcia	Martin	NU
7	12 Munteanu	Adrian	DA
8	13 Nita	Robert	NU
9	14 Stoica	Mihai	NU
10	15 Lopez	Francisc	DA

--- secventa + inserari -> POZITIE

create sequence id_jucator start with 1 increment by 1 minvalue 0 maxvalue 9999 nocycle;

insert into pozitie values(id_pozitie.nextval,'CM','CAM')

insert into pozitie values(id_pozitie.nextval,'CM','CAM')

insert into pozitie values(id_pozitie.nextval, 'LB', 'CB');

insert into pozitie values(id_pozitie.nextval, 'CDM', 'CM');

insert into pozitie values(id_pozitie.nextval, 'CAM', 'CF');

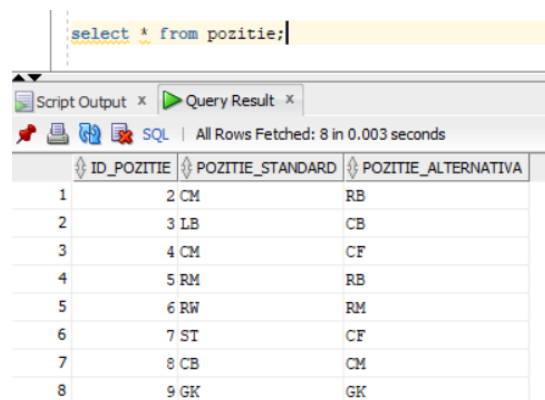
insert into pozitie values(id_pozitie.nextval, 'RM', 'RB');

insert into pozitie values(id_pozitie.nextval, 'RW', 'RM');

insert into pozitie values(id_pozitie.nextval, 'ST', 'CF');

insert into pozitie values(id_pozitie.nextval, 'CB', 'CDM');

insert into pozitie values(id_pozitie.nextval, 'GK', 'GK');



The screenshot shows a SQL query result window with the query `select * from pozitie;` and the following data:

ID_POZITIE	POZITIE_STANDARD	POZITIE_ALTERNATIVA
1	2 CM	RB
2	3 LB	CB
3	4 CM	CF
4	5 RM	RB
5	6 RW	RM
6	7 ST	CF
7	8 CB	CM
8	9 GK	GK

--- secventa + inserari -> JUCATOR

create sequence id_jucator start with 1 increment by 1 minvalue 0 maxvalue 9999 nocycle;

insert into jucator values(id_jucator.nextval, 6, 'Toma', 'David', 2004, 'Romania', 23000000, 92000);

insert into jucator values(id_jucator.nextval, 8, 'Martinez', 'Emi', 1997, 'Argentina', 24000000, 95000);

insert into jucator values(id_jucator.nextval, 8, 'Kim', 'Min-ji', 1998, 'South Korea', 9000000, 78000);

```

insert into jucator values(id_jucator.nextval, 7, 'Ferreira', 'Ricardo', 1993, 'Portugal', 12000000, 89000);
insert into jucator values(id_jucator.nextval, 2, 'Mendoza', 'Javier', 1996, 'Mexico', 33000000, 92000);
insert into jucator values(id_jucator.nextval, 7, 'Bakayoko', 'Tiemoué', 1995, 'France', 11000000, 88000);
insert into jucator values(id_jucator.nextval, 3, 'Vidal', 'Arturo', 1993, 'Chile', 14000000, 93000);
insert into jucator values(id_jucator.nextval, 5, 'Nguyen', 'Minh', 1997, 'Vietnam', 40000000, 80000);
insert into jucator values(id_jucator.nextval, 5, 'Magala', 'Gabriel', 1992, 'Brazil', 13000000, 94000);
insert into jucator values(id_jucator.nextval, 4, 'Lee', 'Soo-jin', 1996, 'South Korea', 115000000, 87000);
insert into jucator values(id_jucator.nextval, 5, 'Andersen', 'Niklas', 2005, 'Denmark', 92000000, 92000);

```

select * from jucator;

ID_JUCATOR	ID_POZITIE	NUME	PRENUME	AN_NASTERE	NATIONALITATE	COTA	SALARIU
1	41	6 Toma	David	2004	Romania	23000000	92000
2	42	8 Martinez	Emi	1997	Argentina	24000000	95000
3	43	8 Kim	Min-ji	1998	South Korea	9000000	78000
4	44	7 Ferreira	Ricardo	1993	Portugal	12000000	89000
5	45	2 Mendoza	Javier	1996	Mexico	33000000	92000
6	46	7 Bakayoko	Tiemoué	1995	France	11000000	88000
7	47	9 Vidal	Arturo	1993	Chile	14000000	93000
8	48	5 Nguyen	Minh	1997	Vietnam	40000000	80000
9	49	3 Magala	Gabriel	1992	Brazil	13000000	94000
10	50	4 Lee	Soo-jin	1996	South Korea	115000000	87000

--- secventa + inserari → STATISTICA

```

create sequence id_statistica start with 1 increment by 1 minvalue 0 maxvalue 9999 nocycle;
insert into statistica values(id_statistica.nextval, 482, 176, 103, 3);
insert into statistica values(id_statistica.nextval, 351, 120, 85, 2);
insert into statistica values(id_statistica.nextval, 264, 68, 42, 1);
insert into statistica values(id_statistica.nextval, 178, 32, 21, 0);
insert into statistica values(id_statistica.nextval, 219, 52, 30, 1);
insert into statistica values(id_statistica.nextval, 97, 19, 10, 0);
insert into statistica values(id_statistica.nextval, 143, 28, 18, 0);
insert into statistica values(id_statistica.nextval, 413, 380, 205, 18);
insert into statistica values(id_statistica.nextval, 140, 100, 50, 2);

```



```
select * from statistica;
```

ID_STATISTICA	NR_MECIURI	NR_GOLURI	NR_ASSISTURI	NR_TROFEE
1	1	482	176	103
2	2	351	120	85
3	3	264	68	42
4	4	178	32	21
5	5	219	52	30
6	6	97	19	10
7	7	143	28	18
8	8	413	380	205
9	9	140	100	50

--- secventa + inserari → ANTRENAMENT

create sequence id_antrenament start with 1 increment by 1 minvalue 0 maxvalue 9999 nocycle;

insert into antrenament values(id_antrenament.nextval, TO_DATE('25-04- 2023','DD-MM-YYYY'), 'fizic')

insert into antrenament values(id_antrenament.nextval, TO_DATE('11-06- 2023','DD-MM-YYYY'), 'fizic')

insert into antrenament values(id_antrenament.nextval, TO_DATE('22-03- 2023','DD-MM-YYYY'), 'tehnica')

insert into antrenament values(id_antrenament.nextval, TO_DATE('05-01- 2023','DD-MM-YYYY'), 'fizic')

insert into antrenament values(id_antrenament.nextval, TO_DATE('14-08- 2023','DD-MM-YYYY'), 'tehnica')

insert into antrenament values(id_antrenament.nextval, TO_DATE('28-02- 2023','DD-MM-YYYY'), 'zi de nastere')

```
select * from antrenament;
```

ID_ANTRENAMENT	DATA	TIP
1	25-APR-23	fizic
2	11-JUN-23	fizic
3	22-MAR-23	tehnica
4	05-JAN-23	fizic
5	14-AUG-23	tehnica
6	28-FEB-23	zi de nastere

--- secventa + inserari -> COMPETITIE

```
create sequence id_competitie start with 1 increment by 1 minvalue 0 maxvalue 9999 nocycle;

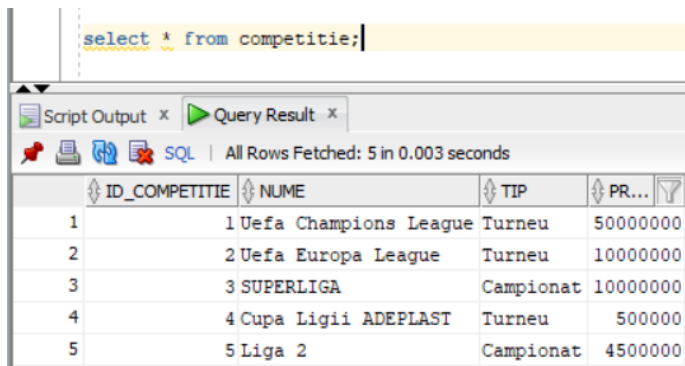
insert into competitie values(id_competitie.nextval,'Uefa Champions League' , 'Turneu' , 50000000);

insert into competitie values(id_competitie.nextval,'Uefa Europa League' , 'Turneu' , 10000000);

insert into competitie values(id_competitie.nextval,'SUPERLIGA' , 'Campionat' , 10000000);

insert into competitie values(id_competitie.nextval,'Cupa Ligii ADEPLAST' , 'Turneu' , 500000);

insert into competitie values(id_competitie.nextval,'Liga 2' , 'Campionat' , 4500000);
```



ID_COMPETITIE	Nume	Tip	PR...
1	1 Uefa Champions League	Turneu	50000000
2	2 Uefa Europa League	Turneu	10000000
3	3 SUPERLIGA	Campionat	10000000
4	4 Cupa Ligii ADEPLAST	Turneu	500000
5	5 Liga 2	Campionat	4500000

--- secventa + inserari -> MECI

```
create sequence id_meci start with 1 increment by 1 minvalue 0 maxvalue 9999 nocycle;

insert into meci values(id_meci.nextval, TO_DATE('03-11- 2022','DD-MM-YYYY'), '1 - 1');

insert into meci values(id_meci.nextval, TO_DATE('11-01- 2023','DD-MM-YYYY'), '1 - 1');

insert into meci values(id_meci.nextval, TO_DATE('14-11- 2022','DD-MM-YYYY'), '3 - 1');

insert into meci values(id_meci.nextval, TO_DATE('06-05- 2023','DD-MM-YYYY'), '5 - 2');

insert into meci values(id_meci.nextval, TO_DATE('26-06- 2011','DD-MM-YYYY'), '1 - 0');

insert into meci values(id_meci.nextval, TO_DATE('08-12- 2002','DD-MM-YYYY'), '17 - 14');

insert into meci values(id_meci.nextval, TO_DATE('02-07- 2014','DD-MM-YYYY'), '7 - 1');
```

select * from meci;

ID_MECI	DATA	SCOR
1	03-NOV-22	1 - 1
2	11-JAN-23	1 - 1
3	14-NOV-22	3 - 1
4	06-MAY-23	5 - 2
5	26-JUN-11	1 - 0
6	08-DEC-02	17 - 14
7	02-JUL-14	7 - 1

--- inserari -> PERFOMEAZA

insert into performeaza values(1,41);

insert into performeaza values(14,42);

insert into performeaza values(13,43);

insert into performeaza values(7,44);

insert into performeaza values(7,46);

insert into performeaza values(2,45);

insert into performeaza values(9,47);

insert into performeaza values(9,50);

insert into performeaza values(9,48);

insert into performeaza values(10,51);

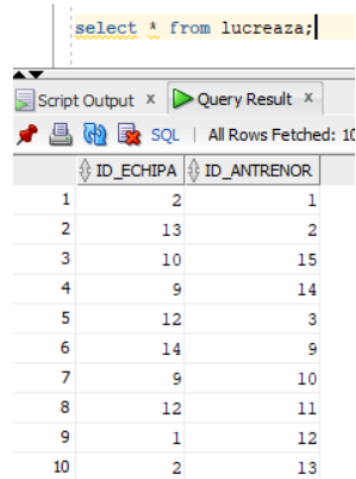
insert into performeaza values(12,49);

select * from performeaza;

ID_ECHIPA	ID_JUCATOR
2	14
3	13
4	7
5	7
6	2
7	9
8	9
9	9
10	10
11	12

--- inserari -> LUCREAZA

```
insert into lucreaza values(2,1);
insert into lucreaza values(13,2);
insert into lucreaza values(12,3);
insert into lucreaza values(14,9);
insert into lucreaza values(9,10);
insert into lucreaza values(12,11);
insert into lucreaza values(1,12);
insert into lucreaza values(2,13);
insert into lucreaza values(9,14);
insert into lucreaza values(10,15);
insert into lucreaza values(10,15);
```

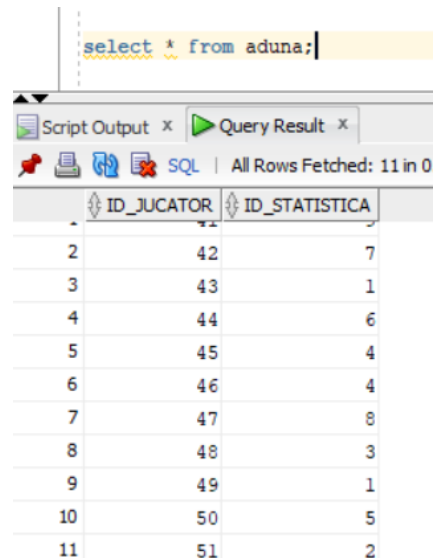


The screenshot shows a SQL query window with the query `select * from lucreaza;` and a results table. The table has two columns: `ID_ECHIPA` and `ID_ANTRENOR`. The results are as follows:

ID_ECHIPA	ID_ANTRENOR
1	2
2	13
3	10
4	9
5	12
6	14
7	9
8	12
9	1
10	2

--- inserari -> ADUNA

```
insert into aduna values(41,9);
insert into aduna values(42,7);
insert into aduna values(43,1);
insert into aduna values(44,6);
insert into aduna values(45,4);
insert into aduna values(46,4);
insert into aduna values(47,8);
insert into aduna values(48,3);
insert into aduna values(49,1);
insert into aduna values(50,5);
insert into aduna values(51,2);
```

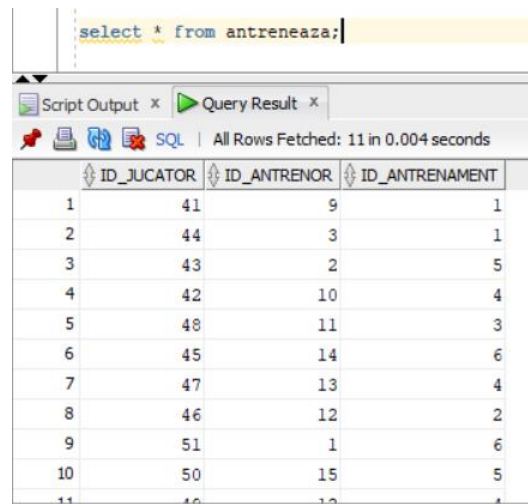


The screenshot shows a SQL query window with the query `select * from aduna;` and a results table. The table has two columns: `ID_JUCATOR` and `ID_STATISTICA`. The results are as follows:

ID_JUCATOR	ID_STATISTICA
2	42
3	43
4	44
5	45
6	46
7	47
8	48
9	49
10	50
11	51

--- inserari -> ANTRENEAZA

```
insert into antreneaza values(41,9,1);
insert into antreneaza values(44,3,1);
insert into antreneaza values(43,2,5);
insert into antreneaza values(42,10,4);
insert into antreneaza values(48,11,3);
insert into antreneaza values(45,14,6);
insert into antreneaza values(47,13,4);
insert into antreneaza values(46,12,2);
insert into antreneaza values(51,1,6);
insert into antreneaza values(50,15,5);
insert into antreneaza values(49,12,4);
```

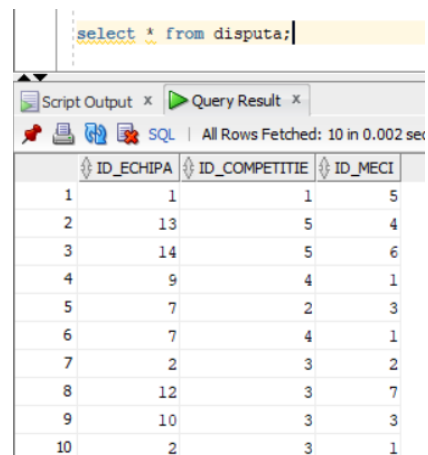


The screenshot shows a SQL query window with the query `select * from antreneaza;` and its results. The results are displayed in a table with 11 rows and 3 columns: ID_JUCATOR, ID_ANTRENOR, and ID_ANTRENAMENT. The status bar indicates 'All Rows Fetched: 11 in 0.004 seconds'.

ID_JUCATOR	ID_ANTRENOR	ID_ANTRENAMENT
1	41	9
2	44	3
3	43	2
4	42	10
5	48	11
6	45	14
7	47	13
8	46	12
9	51	1
10	50	15
11	49	12

--- inserari -> DISPUTA

```
insert into disputa values(1,1,5);
insert into disputa values(13,5,4);
insert into disputa values(14,5,6);
insert into disputa values(9,4,1);
insert into disputa values(7,2,3);
insert into disputa values(7,4,1);
insert into disputa values(2,3,2);
insert into disputa values(12,3,7);
insert into disputa values(2,3,1);
insert into disputa values(10,3,3);
```



The screenshot shows a SQL query window with the query `select * from disputa;` and its results. The results are displayed in a table with 10 rows and 4 columns: ID_ECHIPA, ID_COMPETITIE, ID_MECI, and ID_JUCATOR. The status bar indicates 'All Rows Fetched: 10 in 0.002 seconds'.

ID_ECHIPA	ID_COMPETITIE	ID_MECI	ID_JUCATOR
1	1	1	5
2	13	5	4
3	14	5	6
4	9	4	1
5	7	2	3
6	7	4	1
7	2	3	2
8	12	3	7
9	10	3	3
10	2	3	1

6 - FORMULAȚI ÎN LIMBAJ NATURAL O PROBLEMĂ PE CARE SĂ O REZOLVAȚI FOLOSIND UN SUBPROGRAM STOCAT INDEPENDENT CARE SĂ UTILIZEZE TOATE CELE 3 TIPURI DE COLECȚII STUDIAȚE + APELAȚI SUBPROGRAMUL

Sa se afiseze pentru fiecare echipa jucatorii acesteia, respectiv pentru fiecare jucator, toti antrenorii cu care a interactionat si numarul de antrenamente facut cu fiecare

```
CREATE OR REPLACE PROCEDURE afisare_antrenamente_jucatori IS
```

```
-- Definirea unui RECORD pentru antrenor
```

```
TYPE antrenor_record IS RECORD (
```

```
    id_antrenor antrenor.id_antrenor%TYPE,
```

```
    nume_antrenor antrenor.nume%TYPE,
```

```
    numar_antrenamente NUMBER
```

```
);
```

```
-- Definirea unui VECTOR pentru antrenori
```

```
TYPE vector_antrenori IS VARRAY(200) OF antrenor_record;
```

```
-- Definirea unui RECORD pentru antrenamente
```

```
TYPE antrenamente_record IS RECORD (
```

```
    id_jucator jucator.id_jucator%TYPE,
```

```
    nume_jucator jucator.nume%TYPE,
```

```
    antrenori vector_antrenori
```

```
);
```

```
-- Definirea unui TABLOU IMBRICAT pentru antrenamente
```

```
TYPE antrenamente_nested_table IS TABLE OF antrenamente_record;
```

```
-- Definirea unui TABLOU INDEXAT pentru jucatori cu antrenamente
```

```
TYPE jucatori_antrenamente_indexed_table IS TABLE OF antrenamente_nested_table INDEX BY PLS_INTEGER;
```

```

-- Declararea variabilei pentru stocarea antrenamentelor pentru fiecare echipa
jucatori_cu_antrenamente jucatori_antrenamente_indexed_table;

BEGIN

-- Iterarea prin fiecare echipa
FOR echipa_rec IN (SELECT DISTINCT id echipa, nume echipa FROM echipa)
LOOP

-- Inițializarea antrenamentelor pentru echipa curentă
jucatori_cu_antrenamente(echipa_rec.id echipa) := antrenamente_nested_table();

-- Afișarea informațiilor despre echipa curentă
DBMS_OUTPUT.PUT_LINE('Echipa: ' || echipa_rec.nume echipa);

-- Iterarea prin fiecare jucator din echipa curentă
FOR jucator_rec IN
(SELECT DISTINCT j.id jucator, j.nume || ' ' || j.prenume AS nume_complet
FROM jucator j, performeaza p, echipa e
WHERE j.id jucator = p.id jucator and p.id echipa = echipa_rec.id echipa)
LOOP

-- Inițializarea antrenorilor pentru jucătorul curent
DECLARE
    antrenori vector_antrenori := vector_antrenori();

BEGIN

-- Golește vectorul antrenori
antrenori := vector_antrenori();

-- Iterarea prin fiecare antrenor asociat jucătorului curent
FOR antrenor_rec IN (
    SELECT DISTINCT a.id_antrenor, a.nume || ' ' || a.prenume AS nume_complet, COUNT(*) AS numar_antrenamente
    FROM jucator j, antreneaza an, antrenor a
    WHERE a.id_antrenor = an.id_antrenor AND j.id jucator = jucator_rec.id jucator
    GROUP BY a.id_antrenor, a.nume, a.prenume
)
LOOP

```

```

-- Adăugarea antrenorului la vectorul de antrenori
antrenori.extend();

antrenori(antrenori.last) := antrenor_record(

    id_antrenor => antrenor_rec.id_antrenor,

    nume_antrenor => antrenor_rec.nume_complet,

    numar_antrenamente => antrenor_rec.numar_antrenamente

);

END LOOP;

-- Afișarea informațiilor despre jucătorul curent
DBMS_OUTPUT.PUT_LINE(' Jucator: ' || jucator_rec.nume_complet);

FOR i IN 1..antrenori.count
LOOP
    DBMS_OUTPUT.PUT_LINE(' Antrenor: ' || antrenori(i).id_antrenor ||

        ', Nume: ' || antrenori(i).nume_antrenor ||

        ', Numar antrenamente: ' || antrenori(i).numar_antrenamente);

    END LOOP;

END;

END LOOP;

-- Afișarea delimitatorului între echipe
DBMS_OUTPUT.PUT_LINE('-----');

DBMS_OUTPUT.PUT_LINE('-----');

END LOOP;

END afisare_antrenamente_jucatori;

/

BEGIN

    afisare_antrenamente_jucatori;

END;

/

```

The screenshot shows a SQL IDE interface. The top pane is the 'Query Builder' window, which contains the PL/SQL code from the previous block. The bottom pane is the 'Database Output' window, which displays the results of the execution. The output shows the names of the players and their coaches, separated by a line of dashes.

```

DBMS_OUTPUT.PUT_LINE(' Jucator: ' || jucator_rec.nume_complet);

FOR i IN 1..antrenori.count
LOOP
    DBMS_OUTPUT.PUT_LINE(' Antrenor: ' || antrenori(i).id_antrenor ||
        ', Nume: ' || antrenori(i).nume_antrenor ||
        ', Numar antrenamente: ' || antrenori(i).numar_antrenamente);
    END LOOP;
END;

END LOOP;

-- Afișarea delimitatorului între echipe
DBMS_OUTPUT.PUT_LINE('-----');
DBMS_OUTPUT.PUT_LINE('-----');

END LOOP;

END afisare_antrenamente_jucatori;

/

BEGIN

    afisare_antrenamente_jucatori;

END;

/

```

Database Output:

```

Antrenor: 9, Nume: Brown Michael, Numar antrenamente: 1
Antrenor: 2, Nume: Sumudica Marius, Numar antrenamente: 2
Antrenor: 14, Nume: Stoica Mihai, Numar antrenamente: 1
Jucator: Martinez Emi
Antrenor: 10, Nume: Lopez Alexander, Numar antrenamente: 1
Antrenor: 11, Nume: Garcia Martin, Numar antrenamente: 1

```


7 - FORMULATI IN LIMBAJ NATURAL O PROBLEMĂ PE CARE SĂ O REZOLVAȚI FOLOSIND UN SUBPROGRAM STOCAT INDEPENDENT CARE SĂ UTILIZEZE 2 TIPURI DIFERITE DE CURSOARE STUDIAȚE, UNUL DINTRE ACESTEA FIIND CURSOR PARAMETRIZAT, DEPENDENT DE CELĂLALT CURSOR. APELAȚI SUBPROGRAMUL.

Pentru fiecare echipă din Italia, afișați jucătorii care au evoluat în echipă, împreună cu statisticile fiecăruia, incluzând numărul de meciuri în care au participat și numărul total de contribuții (goluri + asisturi).

```
CREATE OR REPLACE PROCEDURE afisare_statistici_jucatori IS

-- Declarația cursorului explicit

CURSOR echipe_italia_cursor IS

    SELECT id echipa, nume echipa

    FROM echipa

    WHERE nume echipa IN ('AC Milan', 'Inter Milano');

-- Declarația cursorului parametrizat

CURSOR statistici_cursor (id echipa_param IN echipa.id echipa%TYPE) IS

    SELECT j.nume || ' ' || j.prenume AS nume_jucator,

           s.nr_meciuri AS numar_meciuri,

           SUM(s.nr_goluri + s.nr_asisturi) AS total_contributii

    FROM jucator j

    JOIN performeaza p ON j.id_jucator = p.id_jucator

    JOIN echipa e ON e.id echipa = p.id echipa

    JOIN aduna a ON j.id_jucator = a.id_jucator

    JOIN statistica s ON s.id_statistica = a.id_statistica

    WHERE e.id echipa = id echipa_param
```

```
GROUP BY j.numa, j.prenume, s.nr_meciuri;
```

```
-- Variabile pentru a stoca datele
```

```
v_id echipa echipa.id_echipa%TYPE;
```

```
v_numa echipa echipa.numa_echipa%TYPE;
```

```
v_numa_jucator VARCHAR2(200);
```

```
v_numar_meciuri NUMBER;
```

```
v_total_contributii NUMBER;
```

```
BEGIN
```

```
-- Deschidem cursorul echipei din Italia
```

```
OPEN echipe_italia_cursor;
```

```
LOOP
```

```
FETCH echipe_italia_cursor INTO v_id_echipa, v_numa_echipa;
```

```
EXIT WHEN echipe_italia_cursor%NOTFOUND;
```

```
DBMS_OUTPUT.PUT_LINE('Echipa: ' || v_numa_echipa);
```

```
-- Deschidem cursorul parametrizat pentru echipa curentă
```

```
OPEN statistici_cursor(v_id_echipa);
```

```
LOOP
```

```
FETCH statistici_cursor INTO v_numa_jucator, v_numar_meciuri, v_total_contributii;
```

```
EXIT WHEN statistici_cursor%NOTFOUND;
```

```
DBMS_OUTPUT.PUT_LINE(' Jucator: ' || v_numa_jucator ||
```

```
 ', Meciuiri: ' || v_numar_meciuri ||
```

```
 ', Total Contributii (Goluri + Asisturi): ' || v_total_contributii);
```

END LOOP;

-- Închidem cursorul parametrizat

CLOSE statistici_cursor;

END LOOP;

-- Închidem cursorul echipei din Italia

CLOSE echipe_italia_cursor;

END afisare_statistici_jucatori;

/

-- Apelarea subprogramului

BEGIN

afisare_statistici_jucatori;

END;

/

The screenshot displays the Oracle SQL Developer interface. The top pane, titled 'Query Builder', contains the following PL/SQL code:

```
END afisare_statistici_jucatori;
/

-- Apelarea subprogramului
BEGIN
    afisare_statistici_jucatori;
END;
/
```

The bottom pane, titled 'Script Output', shows the execution results of the procedure. It lists the team and player statistics for three teams: AC Milan, Inter Milano, and Inter Milano. The results are as follows:

Echipa	Jucator	Meciuri	Total Contributii (Goluri + Asisturi)
AC Milan	Ferreira Ricardo	178	53
	Bakayoko Tiemoué	413	585
Inter Milano	Nguyen Minh	264	110

Below the table, the message 'PL/SQL procedure successfully completed.' is displayed.

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 SQL 3 dintre tabelele definite.

Definiți minim 2 excepții proprii. Apelați subprogramul astfel încât să evidențiați toate cazurile definite și tratate.

Pentru un jucator dat dupa nume, sa se afiseze numarul de antrenamente facute in subordinea unui antrenor licentiate.

```
CREATE OR REPLACE FUNCTION numar_antrenamente_licentiate
(par_nume_jucator IN jucator.nume%TYPE)
RETURN NUMBER
IS
v_numar_antrenamente NUMBER := 0;
v_id_jucator jucator.id_jucator%TYPE;
BEGIN
-- Selectarea numărului de antrenamente sub antrenori licențiați
BEGIN
SELECT j.id_jucator, COUNT(*)
INTO v_id_jucator, v_numar_antrenamente
FROM jucator j
JOIN antreneaza a ON j.id_jucator = a.id_jucator
JOIN antrenament antr ON antr.id_antrenament = a.id_antrenament
JOIN antrenor an ON a.id_antrenor = an.id_antrenor
WHERE INITCAP(j.nume) = INITCAP(par_nume_jucator)
AND an.licenta = 'DA'
GROUP BY j.id_jucator;

-- Verificarea dacă numărul de antrenamente este 0
IF v_numar_antrenamente = 0 THEN
RAISE_APPLICATION_ERROR(-20000, 'Nu există antrenamente pentru jucătorul cu numele ' || par_nume_jucator);
```

```

END IF;

EXCEPTION

WHEN NO_DATA_FOUND THEN

    RAISE_APPLICATION_ERROR(-20000, 'Nu există niciun jucător cu numele ' || par_nume_jucator);

WHEN TOO_MANY_ROWS THEN

    RAISE_APPLICATION_ERROR(-20001, 'Există mai mulți jucători cu numele ' || par_nume_jucator);

WHEN OTHERS THEN

    RAISE_APPLICATION_ERROR(-20002, 'A apărut o eroare neașteptată: ' || SQLERRM);

END;

RETURN v_numar_antrenamente;

```

END numar_antrenamente_licentiate;

```

SELECT numar_antrenamente_licentiate('Vidal') FROM DUAL;
SELECT numar_antrenamente_licentiate('Kim') FROM DUAL; -- mai multi cu acelasi nume
SELECT numar_antrenamente_licentiate('Roberto') FROM DUAL; -- nu exista

```

NUMAR_ANTRENAMENTE_LICENTIAȚE(VIDAL)	
1	1

```

ORA-20000: Nu există niciun jucător cu numele Roberto
ORA-06512: at "UTILIZATOR.NUMAR_ANTRENAMENTE_LICENTIAȚE", line 26
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.

```

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

Pentru o echipă specifică, să se calculeze și să se afișeze numărul total de trofee câștigate de jucătorii echipei respective.

```
CREATE OR REPLACE PROCEDURE palmares_individual
(par_num_echipa IN echipa.num_echipa%TYPE)
IS
v_id_echipa echipa.id_echipa%TYPE;
v_num_echipa VARCHAR(20);
v_total_trofee NUMBER;
BEGIN
    BEGIN
        -- Calcularea numărului total de trofee câștigate de jucătorii echipei specificate
        SELECT e.id_echipa, e.num_echipa, NVL(SUM(s.nr_trofee), 0)
        INTO v_id_echipa, v_num_echipa, v_total_trofee
        FROM echipa e
        JOIN performeaza p ON e.id_echipa = p.id_echipa
        JOIN jucator j ON p.id_jucator = j.id_jucator
        JOIN aduna a ON j.id_jucator = a.id_jucator
        JOIN statistica s ON a.id_statistica = s.id_statistica
        WHERE INITCAP(e.num_echipa) = INITCAP(par_num_echipa)
        GROUP BY e.id_echipa, e.num_echipa;

        -- Afișarea rezultatului
        DBMS_OUTPUT.PUT_LINE(v_num_echipa || ' - ' || v_total_trofee || ' trofee');
```

EXCEPTION

WHEN NO_DATA_FOUND THEN

DBMS_OUTPUT.PUT_LINE('Nu exista echipa cu numele ' || par_nume_echipa);

WHEN TOO_MANY_ROWS THEN

DBMS_OUTPUT.PUT_LINE('Exista mai multe echipe cu același nume ' || par_nume_echipa);

WHEN OTHERS THEN

DBMS_OUTPUT.PUT_LINE('A aparut o eroare neasteptata: ' || SQLERRM);

END;

END;

/

BEGIN

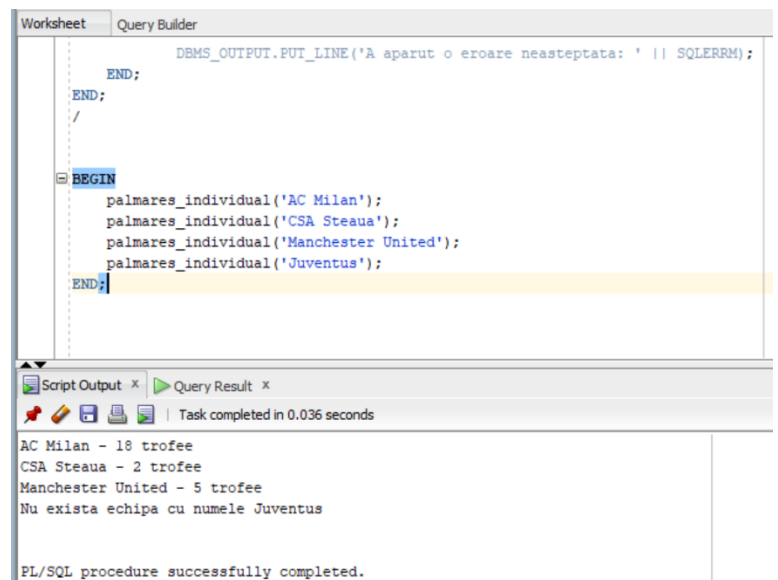
palmares_individual('AC Milan');

palmares_individual('CSA Steaua');

palmares_individual('Manchester United');

palmares_individual('Juventus');

END;



```
Worksheet Query Builder
DBMS_OUTPUT.PUT_LINE('A aparut o eroare neasteptata: ' || SQLERRM);
END;
END;
/

BEGIN
palmares_individual('AC Milan');
palmares_individual('CSA Steaua');
palmares_individual('Manchester United');
palmares_individual('Juventus');
END;

Script Output x Query Result x
Task completed in 0.036 seconds
AC Milan - 18 trofee
CSA Steaua - 2 trofee
Manchester United - 5 trofee
Nu exista echipa cu numele Juventus
PL/SQL procedure successfully completed.
```

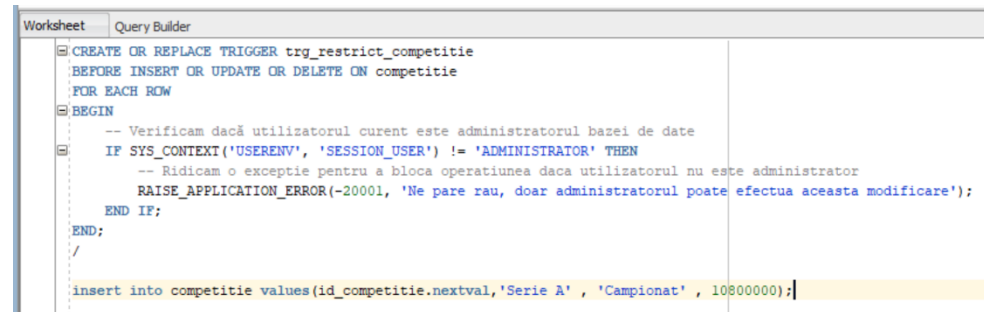
MENTIUNE: Nu am eroarea de “TOO MANY ROWS” pentru ca am deja constrangere pe tabelul “echipa” ca numele acestora sa fie distincte

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

Implementati un trigger prin care doar administratorul bazei de date poate modificari (INSERT, UPDATE, DELETE) asupra unei competitii din baza de date.

```
CREATE OR REPLACE TRIGGER trg_restrict_competitie
BEFORE INSERT OR UPDATE OR DELETE ON competitie
FOR EACH ROW
BEGIN
    -- Verificam dacă utilizatorul curent este administratorul bazei de date
    IF SYS_CONTEXT('USERENV', 'SESSION_USER') != 'ADMINISTRATOR' THEN
        -- Ridicam o exceptie pentru a bloca operatiunea daca utilizatorul nu este administrator
        RAISE_APPLICATION_ERROR(-20001, 'Ne pare rau, doar administratorul poate efectua aceasta modificare');
    END IF;
END;
```

/



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

Implementati un trigger care sa nu permita unui jucator sa aiba salariul mai mare de 128.000

```
CREATE OR REPLACE TRIGGER trg_fair_play_financiar
BEFORE INSERT OR UPDATE ON jucator
FOR EACH ROW
```


BEGIN

-- Verificam dacă salariul este mai mare decât 128.000

IF :NEW.salariu > 128000 THEN

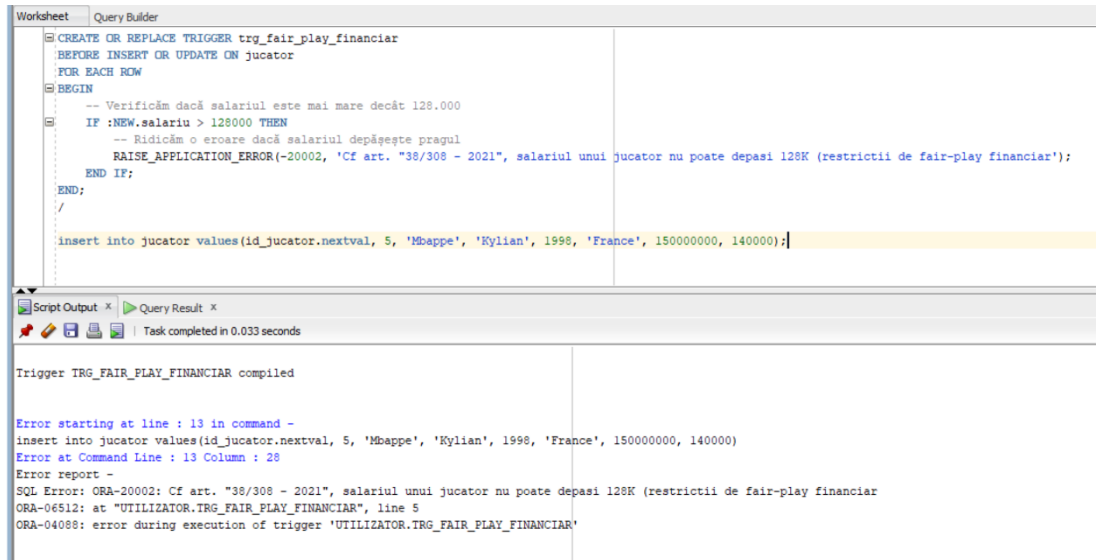
-- Ridicam o eroare dacă salariul depășește pragul

RAISE_APPLICATION_ERROR(-20002, 'Cf art. "38/308 - 2021", salariul unui jucator nu poate depasi 128K (restrictii de fair-play financiar)');

END IF;

END;

/



The screenshot shows the SQL Developer interface. The top pane displays the SQL code for creating a trigger named `trg_fair_play_financiar`. The trigger is set to fire before insert or update on the `jucator` table. The code includes a comment in Romanian about verifying the salary and raising an application error if it exceeds 128,000. Below the trigger code, an `insert` statement is shown, attempting to insert a player with a salary of 1,500,000,000. The bottom pane shows the 'Script Output' window, which reports that the trigger compiled successfully but then shows an error starting at line 13 in the command. The error message is: 'ORA-20002: Cf art. "38/308 - 2021", salariul unui jucator nu poate depasi 128K (restrictii de fair-play financiar)'. The error also mentions 'ORA-06512: at "UTILIZATOR.TRG_FAIR_PLAY_FINANCIAR", line 5' and 'ORA-04088: error during execution of trigger "UTILIZATOR.TRG_FAIR_PLAY_FINANCIAR"'.

```
CREATE OR REPLACE TRIGGER trg_fair_play_financiar
BEFORE INSERT OR UPDATE ON jucator
FOR EACH ROW
BEGIN
    -- Verificăm dacă salariul este mai mare decât 128.000
    IF :NEW.salariu > 128000 THEN
        -- Ridicăm o eroare dacă salariul depășește pragul
        RAISE_APPLICATION_ERROR(-20002, 'Cf art. "38/308 - 2021", salariul unui jucator nu poate depasi 128K (restrictii de fair-play financiar)');
    END IF;
END;
/

insert into jucator values(id_jucator.nextval, 5, 'Mbappe', 'Kylian', 1998, 'France', 1500000000, 140000);
```

Trigger TRG_FAIR_PLAY_FINANCIAR compiled

Error starting at line : 13 in command -
insert into jucator values(id_jucator.nextval, 5, 'Mbappe', 'Kylian', 1998, 'France', 1500000000, 140000)
Error at Command Line : 13 Column : 28
Error report -
SQL Error: ORA-20002: Cf art. "38/308 - 2021", salariul unui jucator nu poate depasi 128K (restrictii de fair-play financiar)
ORA-06512: at "UTILIZATOR.TRG_FAIR_PLAY_FINANCIAR", line 5
ORA-04088: error during execution of trigger 'UTILIZATOR.TRG_FAIR_PLAY_FINANCIAR'

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

Implementati un trigger care sa nu permita modificari (CREATE, ALTER, DROP) decat intre orele 10 si 18

CREATE OR REPLACE TRIGGER trg_restrict_modifications

BEFORE CREATE OR ALTER OR DROP ON SCHEMA

DECLARE

v_current_hour NUMBER;

BEGIN

-- Obtine ora curenta

v_current_hour := TO_NUMBER(TO_CHAR(SYSDATE, 'HH24'));

-- Verifica dacă ora curenta este în afara intervalului permis

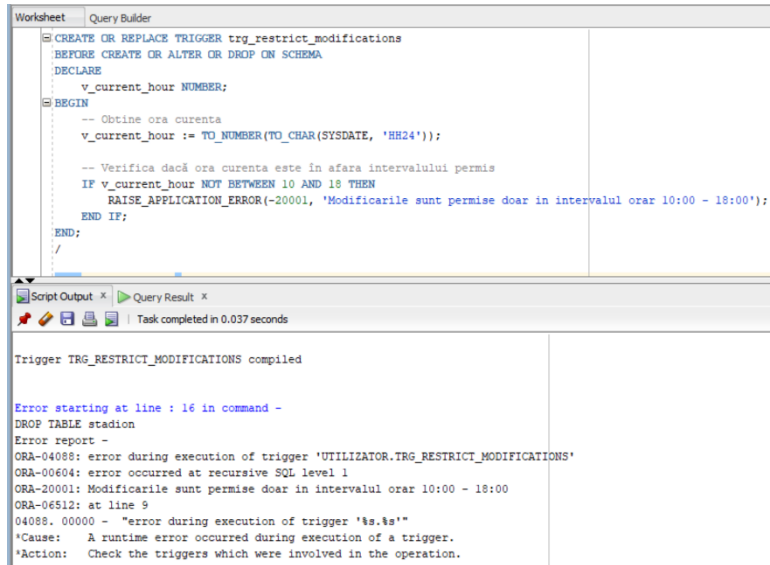
IF v_current_hour NOT BETWEEN 10 AND 18 THEN

RAISE_APPLICATION_ERROR(-20001, 'Modificarile sunt permise doar in intervalul orar 10:00 - 18:00');

END IF;

END;

/



The screenshot displays the SQL Developer interface. The top pane, titled 'Query Builder', contains the following SQL script:

```
CREATE OR REPLACE TRIGGER trg_restrict_modifications
BEFORE CREATE OR ALTER OR DROP ON SCHEMA
DECLARE
    v_current_hour NUMBER;
BEGIN
    -- Obtine ora curenta
    v_current_hour := TO_NUMBER(TO_CHAR(SYSDATE, 'HH24'));

    -- Verifica dacă ora curenta este în afara intervalului permis
    IF v_current_hour NOT BETWEEN 10 AND 18 THEN
        RAISE_APPLICATION_ERROR(-20001, 'Modificarile sunt permise doar in intervalul orar 10:00 - 18:00');
    END IF;
END;
```

The bottom pane shows the 'Script Output' and 'Query Result' tabs. The 'Script Output' tab is active, displaying the following messages:

```
Trigger TRG_RESTRICT_MODIFICATIONS compiled

Error starting at line : 16 in command -
DROP TABLE stadion
Error report -
ORA-04088: error during execution of trigger 'UTILIZATOR.TRG_RESTRICT_MODIFICATIONS'
ORA-00604: error occurred at recursive SQL level 1
ORA-20001: Modificarile sunt permise doar in intervalul orar 10:00 - 18:00
ORA-06512: at line 9
04088. 00000 - "error during execution of trigger '%s.%s'"
*Cause:      A runtime error occurred during execution of a trigger.
*Action:     Check the triggers which were involved in the operation.
```