Gestionarea unui club de biliard

1.

Voi prezenta in prima parte a proiectului modelul bazei de date, utilitatea, restrictiile si regulile de funtionare ale acestuia.

Tema proiectului meu este proiectarea bazei de date a unui club de biliard. Aceasta contine informatii cu privire la jucatorii si antrenorii unui club de biliard, care sunt gestionati de catre un manager. Jucatorii se antreneaza intr-o sala cu cel putin o masa de biliard, salile fiind intretinute de catre un administrator. Jucatorii pot participa, pe parcursul unui sezon, la competitii de biliard, unde au ocazia sa castige premii oferite de sponsori (cel putin un sponsor).

Scopul creării acestei baze de date este pentru a putea tine evidenta jucatorilor de biliard ai clubului, care sunt antrenati pentru a participa la competitii.

Fiecare jucator participa la competitii (cel putin una pe sezon), unde disputa meciuri (cel putin un meci) contra altor jucatori (cel putin un jucator). In functie de locul pe care acesta se claseaza la finalul competitiei castiga un premiu (exista cel putin un premiu in fiecare competitie).

2.

Modelul de date respecta urmatoarele restrictii de functionare:

Clubul are un singur manager care are angajati cel putin un jucator si un antrenor.

Un antrenor antreneaza cel putin un jucator.

La un antrenament se prezinta cel putin un antrenor.

Un antrenament se desfasoara la o singura masa de biliard.

O sala contine cel putin o masa de biliard.

Un administrator se ocupa (administreaza) cel putin o sala de acest fel.

Cel putin un jucator participa la cel putin un meci pe sezon, respectiv la cel putin o competitie pe sezon.

La minim un meci, respectiv la minim o competitie, trebuie sa se prezinte cel putin un jucator, pentru a se putea determina un castigator.

Se organizeaza cel putin o competitie pe sezon, iar fiecare dintre acestea contine minim un meci.

Un sezon de biliard contine cel putin un meci si cel putin o competitie.

O competitie este organizata de cel putin un organizator si se desfasoara in cel putin o sala.

In cadrul unei competitii se acorda cel putin un premiu.

Cel putin un premiu (de obicei mai multe premii) este oferit de catre cel putin un sponsor.

3.

Modelul bazei de date a unui club de biliard contine urmatoarele entitati:

**Manager** – persoana care se ocupa de gestionarea angajatilor clubului (jucatori si antrenori).

Cheie primara: id\_manager

**Antrenor** – persoana care organizeaza antrenamentele jucatorilor si are grija de sanatatea fizica si mentala a acestora.

Cheie primara: id\_antrenor

**Jucator** – persoana care se ocupa cu practicarea biliardului la nivel de performanta, participand la competitii si meciuri.

Cheie primara: id\_jucator

**Antrenament** – perioada (de obicei cu durata de 1-2 ore) pe parcursul careia jucatorul se pregateste fizic si mental pentru meciuri si isi imbunatateste calitatile tehnice si fizice.

Cheie primara: id\_antrenament

**Masa** – suprafata de joc, speciala pentru desfasurarea jocurilor de biliard.

Cheie primara: id\_masa

**Sala** – o locatie interioara care contine mai multe mese, care are, de obicei, un cost de inchiriere.

Cheie primara: id\_sala

**Administrator** – persoana care se ocupa de intretinerea intr-o stare buna sau foarte buna a salilor si de contractele incheiate pentru inchirierea acestora.

Cheie primara: id\_admin

**Meci** – partida intre doi jucatori, care consta in desfasurarea mai multor jocuri de biliard, avand in final un singur castigator.

Cheie primara: id\_meci

**Sezon**: perioada de-a lungul careia se desfasoara meciuri si competitii, la finalul careia fiecare jucator ocupa un anumit loc in clasament (de obicei debuteaza in toamna si se incheie odata cu inceputul verii).

Cheie primara: id\_sezon

**Competitie** – concurs care se desfasoară în mai multe etape, pe parcursul unei perioade mai lungi de timp, pentru obtinerea unui titlu sau a unei performante sportive.

Cheie primara: id\_comp

**Organizator** – persoana sau grup de persoane (firma) care se ocupa cu organizarea si buna desfasurare a unei competitii.

Cheie primara: id\_org

**Premiu** – recompensa acordata castigatorului unei competitii sau a ocupantului unui anumit loc in clasament (de obicei se acorda un premiu, reprezentat de o suma de bani, fiecarui participant al competitiei dar, pe masura ce jucatorul avanseaza etape, premiul creste exponential; cei care nu trec de primele etape castiga premii mici).

Cheie primara: id\_premiu

**Sponsor** - persoană fizică sau juridică având calitatea de a sponsoriza o competitie.

Cheie primara: id\_sponsor

**Participa** – actiune prin care un jucator ia parte la un meci, respectiv o competitie.

Chei primare: id\_jucator, id\_meci, id\_comp

**Ofera** – a fi pus la dispozitie unul sau mai multe premii (de catre cel putin un sponsor), pentru a fi oferit jucatorilor aflati pe pozitii castigatoare in clasament.

Chei primare: id\_sponsor, id\_premiu

4.

Relatiile modelului de date sunt urmatoarele:

Manager angajeaza Antrenor - relatie de tip one to many, cardinalitate minima 1:1, cardinalitate maxima 1:n.

Manager gestineaza Jucator - relatie de tip one to many, cardinalitate minima 1:1, cardinalitate maxima 1:n.

Antrenor antreneaza Jucator - relatie de tip one to many, cardinalitate minima 1:1, cardinalitate maxima 1:n.

Jucator se\_antreneaza\_la antrenament - relatie de tip one to many, cardinalitate minima 1:1, cardinalitate maxima 1:n.

Antrenor organizeaza Antrenament - relatie de tip one to many, cardinalitate minima 1:1, cardinalitate maxima 1:n.

Antrenament se\_desfasoara\_la Masa - relatie de tip one to one, cardinalitate minima 1:1, cardinalitate maxima 1:1.

Sala contine Masa - relatie de tip one to many, cardinalitate minima 1:1, cardinalitate maxima 1:n.

Administrator administreaza Sala - relatie de tip one to many, cardinalitate minima 1:1, cardinalitate maxima 1:n.

Jucator participa\_la Meci si\_la competitie - relatie de tip 3, ce leaga entitatile Jucator, Meci si Competitie.

Sezon contine Meci - relatie de tip one to many, cardinalitate minima 1:1, cardinalitate maxima 1:n.

Sezon contine Competitie - relatie de tip one to many, cardinalitate minima 1:1, cardinalitate maxima 1:n.

Competitie este\_organizata\_de Organizator - relatie de tip one to many, cardinalitate minima 1:1, cardinalitate maxima 1:n.

Competitia are Premiu - relatie de tip one to many, cardinalitate minima 1:1, cardinalitate maxima 1:n.

Sponsor ofera Premiu - relatie de tip many to many, cardinalitate minima 1:1, cardinalitate maxima m:n.

Competitie se\_desfasoara\_in Sala - relatie de tip one to many, cardinalitate minima 1:1, cardinalitate maxima 1:n.

5.

Entitatea Manager are ca atribute:

id\_manager – codul managerului, variabila de tip intreg de lungime maxima 6, constrangere: PRIMARY KEY

nume\_manager - numele managerului, variabila de tip sir de caractere de lungime maxima 100, constrangere: NOT NULL

telefon\_manager – numarul de telefon al managerului, variabila de tip intreg de lungime 10

Entitatea Antrenor are ca atribute:

id\_antrenor – codul antrenorului, variabila de tip intreg de lungime maxima 6, constrangere: PRIMARY KEY

nume\_ antrenor - numele antrenorului, variabila de tip sir de caractere de lungime maxima 100, constrangere: NOT NULL

nr\_jucatori – numarul de jucatori pe care ii antreneaza, variabila de tip intreg de lungime maxima 3, constrangere: NOT NULL

salariu\_antrenor – salariul antrenorului, exprimat in lei, variabila de tip intreg de lungime maxima 10

Entitatea Jucator are ca atribute:

id\_ jucator – codul jucatorului, variabila de tip intreg de lungime maxima 6, constrangere: PRIMARY KEY

nume\_ jucator - numele jucatorului, variabila de tip sir de caractere de lungime maxima 100, constrangere: NOT NULL

varsta\_ jucator – varsta jucatorului, variabila de tip intreg de lungime maxima 3, constrangere: NOT NULL

loc\_clasament – locul pe care se afla jucatorul in clasamentul mondial, variabila de tip intreg de lungime maxima 10, constrangere: NOT NULL

salariu\_jucator – salariul jucatorului, exprimat in lei, variabila de tip intreg de lungime maxima 10

Entitatea Antrenament are ca atribute:

id\_ antrenament – codul antrenamentului, variabila de tip intreg de lungime maxima 6, constrangere: PRIMARY KEY

ora\_ antrenament – ora antrenamentului (antrenamentul incepe doar la ora fixa), variabila de tip intreg de lungime maxima 2, constrangere: NOT NULL

durata\_ antrenament – durata antrenamentului, exprimata in minute, variabila de tip intreg de lungime maxima 3, constrangere: NOT NULL

Entitatea Masa are ca atribute:

id\_ masa – codul mesei, variabila de tip intreg de lungime maxima 6, constrangere: PRIMARY KEY

pret masa – pretul mesei, exprimat in lei, variabila de tip intreg de lungime maxima 10

Entitatea Sala are ca atribute:

id\_ sala – codul salii, variabila de tip intreg de lungime maxima 6, constrangere: PRIMARY KEY

locatie – locatia salii, exprimata prin numele strazii urmat de numarul acesteia, variabila de tip sir de caractere de lungime maxima 100, constrangere: NOT NULL

nr\_mese – numarul de mese din sala, variabila de tip intreg de lungime maxima 2, constrangere: NOT NULL

pret\_inchiriere – pretul de inchiriere al salii pe luna, exprimat in lei, variabila de tip intreg de lungime maxima 10

Entitatea Administrator are ca atribute:

id\_admin – codul administratorului, variabila de tip intreg de lungime maxima 6, constrangere: PRIMARY KEY

nume\_admin - numele administratorului, variabila de tip sir de caractere de lungime maxima 100, constrangere: NOT NULL

telefon\_admin – numarul de telefon al administratorului, variabila de tip intreg de lungime 10

Entitatea Meci are ca atribute:

id\_ meci – codul meciului, variabila de tip intreg de lungime maxima 6, constrangere: PRIMARY KEY

jucator1 - numele primului jucator din meci, variabila de tip sir de caractere de lungime maxima 100, constrangere: NOT NULL

jucator2 - numele celui de-al doilea jucator din meci, variabila de tip sir de caractere de lungime maxima 100, constrangere: NOT NULL

nr\_jocuri – numarul de jocuri disputate pana la incheierea meciului, variabila de tip intreg de lungime maxima 2 (poate fi NULL, in cazul in care un jucator nu se prezinta la meci)

Entitatea Sezon are ca atribute:

id\_ sezon – codul sezonului, variabila de tip intreg de lungime maxima 6, constrangere: PRIMARY KEY

data\_incep – data inceperii sezonului, variabila de tip data calendaristica, constrangere: NOT NULL

data\_fin – data finalizarii sezonului, variabila de tip data calendaristica, constrangere: NOT NULL

Entitatea Competitie are ca atribute:

id\_ comp – codul competitiei, variabila de tip intreg de lungime maxima 6, constrangere: PRIMARY KEY

data\_comp – data inceperii competitiei, variabila de tip data calendaristica, constrangere: NOT NULL

nr\_competitori – numarul de competitori care participa la competitie, variabila de tip intreg de lungime maxima 5, constrangere: NOT NULL

nume\_comp - numele competitiei, variabila de tip sir de caractere de lungime maxima 100, constrangere: NOT NULL

Entitatea Organizator are ca atribute:

id\_org – codul organizatorului sau al firmei organizatoare, variabila de tip intreg de lungime maxima 6, constrangere: PRIMARY KEY

nume\_org - numele organizatorului sau al firmei organizatoare, variabila de tip sir de caractere de lungime maxima 100, constrangere: NOT NULL

telefon\_org – numarul de telefon al organizatorului sau al firmei organizatoare, variabila de tip intreg de lungime 10, constrangere: NOT NULL

Entitatea Premiu are ca atribute:

id\_premiu – codul premiului, variabila de tip intreg de lungime maxima 6, constrangere: PRIMARY KEY

clasare – pozitia pe care jucatorul trebuie sa incheie competitia pentru a incasa premiul, variabila de tip intreg de lungime maxima 3 (poate fi NULL in cazul in care jucatorul termina pe o pozitie care nu ofera niciun premiu)

valoare – valoarea premiului, exprimata in lei, variabila de tip intreg de lungime maxima 10, constrangere: NOT NULL

Entitatea Sponsor are ca atribute:

id\_sponsor – codul sponsorului, variabila de tip intreg de lungime maxima 6, constrangere: PRIMARY KEY

nume\_sponsor - numele sponsorului, variabila de tip sir de caractere de lungime maxima 100, constrangere: NOT NULL

Entitatea Participa are ca atribute:

id\_ jucator – codul jucatorului, variabila de tip intreg de lungime maxima 6, constrangere: PRIMARY KEY

id\_ meci – codul meciului, variabila de tip intreg de lungime maxima 6, constrangere: PRIMARY KEY

id\_ comp – codul competitiei, variabila de tip intreg de lungime maxima 6, constrangere: PRIMARY KEY

rezultat – numele jucatorului castigator al etapei respective din competitie, variabila de tip sir de caractere de lungime maxima 100, constrangere: NOT NULL

nr\_victorii – numarul de victorii din cadrul competitiei, al jucatorului, variabila de tip intreg de lungime maxima 2

Entitatea Ofera are ca atribute:

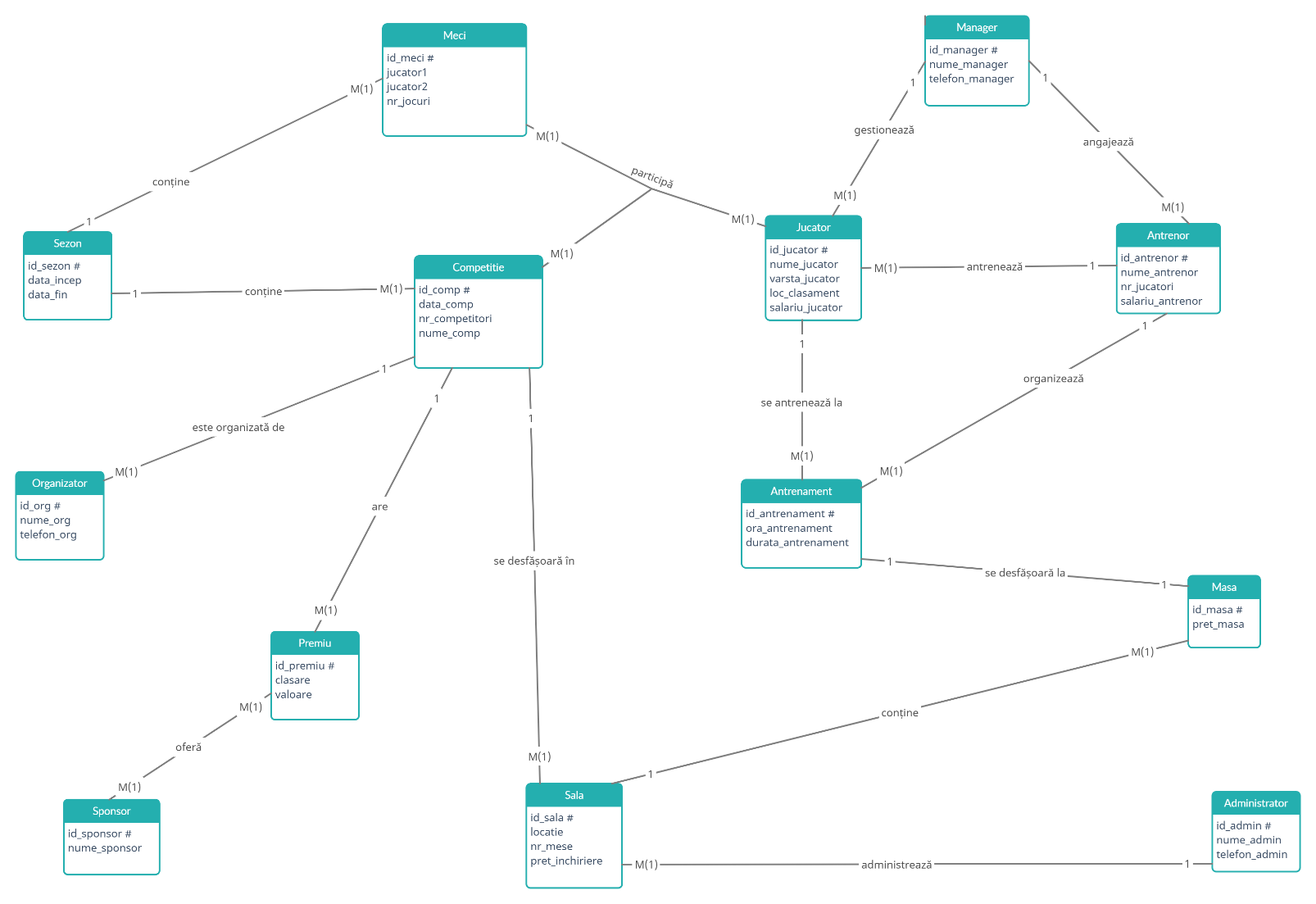
id\_sponsor – codul sponsorului, variabila de tip intreg de lungime maxima 6, constrangere: PRIMARY KEY

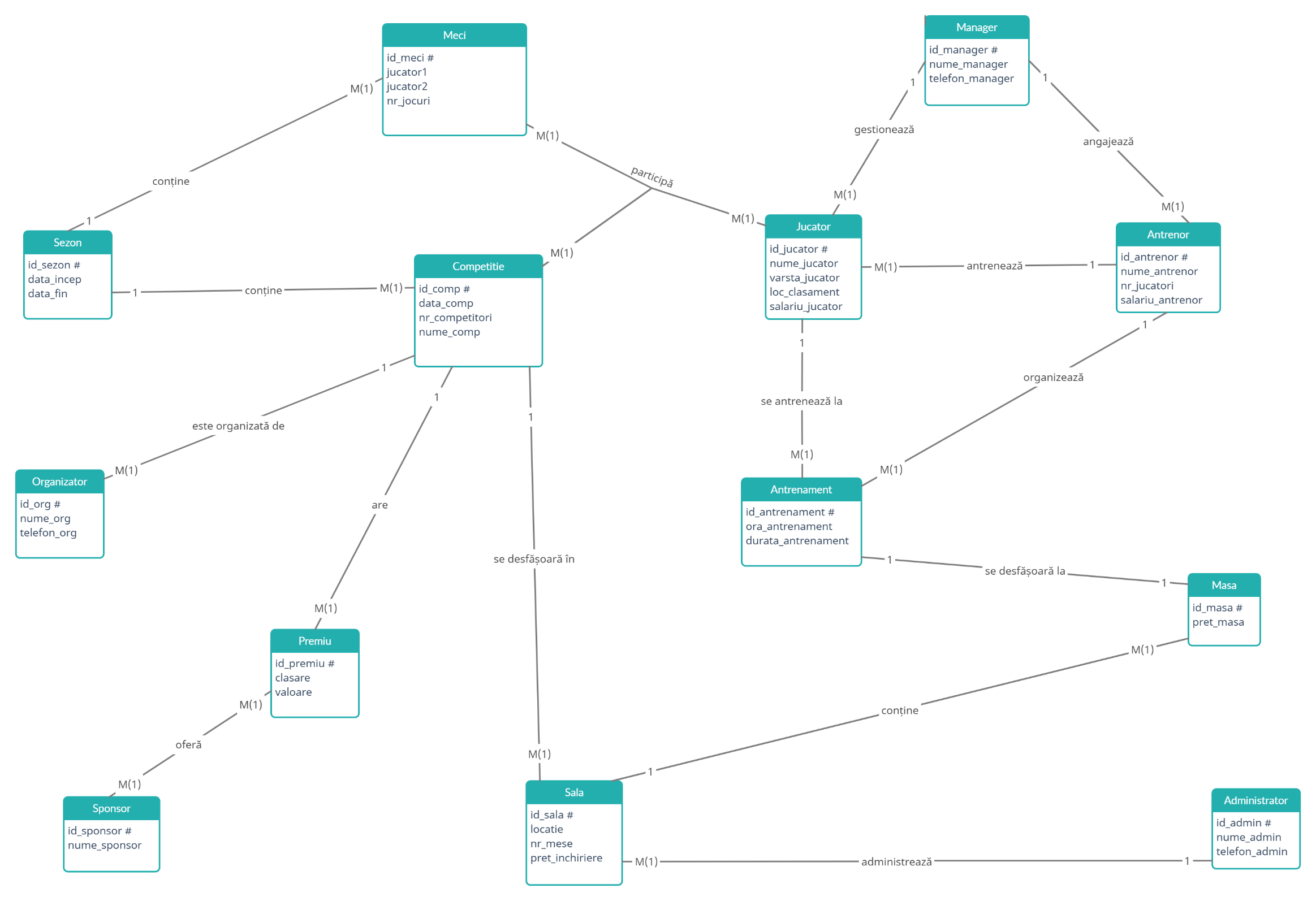
id\_premiu – codul premiului, variabila de tip intreg de lungime maxima 6, constrangere: PRIMARY KEY

suma\_totala – suma totala, exprimata in lei, investita de catre sponsor(i) in premiu/premii, variabila de tip intreg de lungime maxima 10, constrangere: NOT NULL

6.

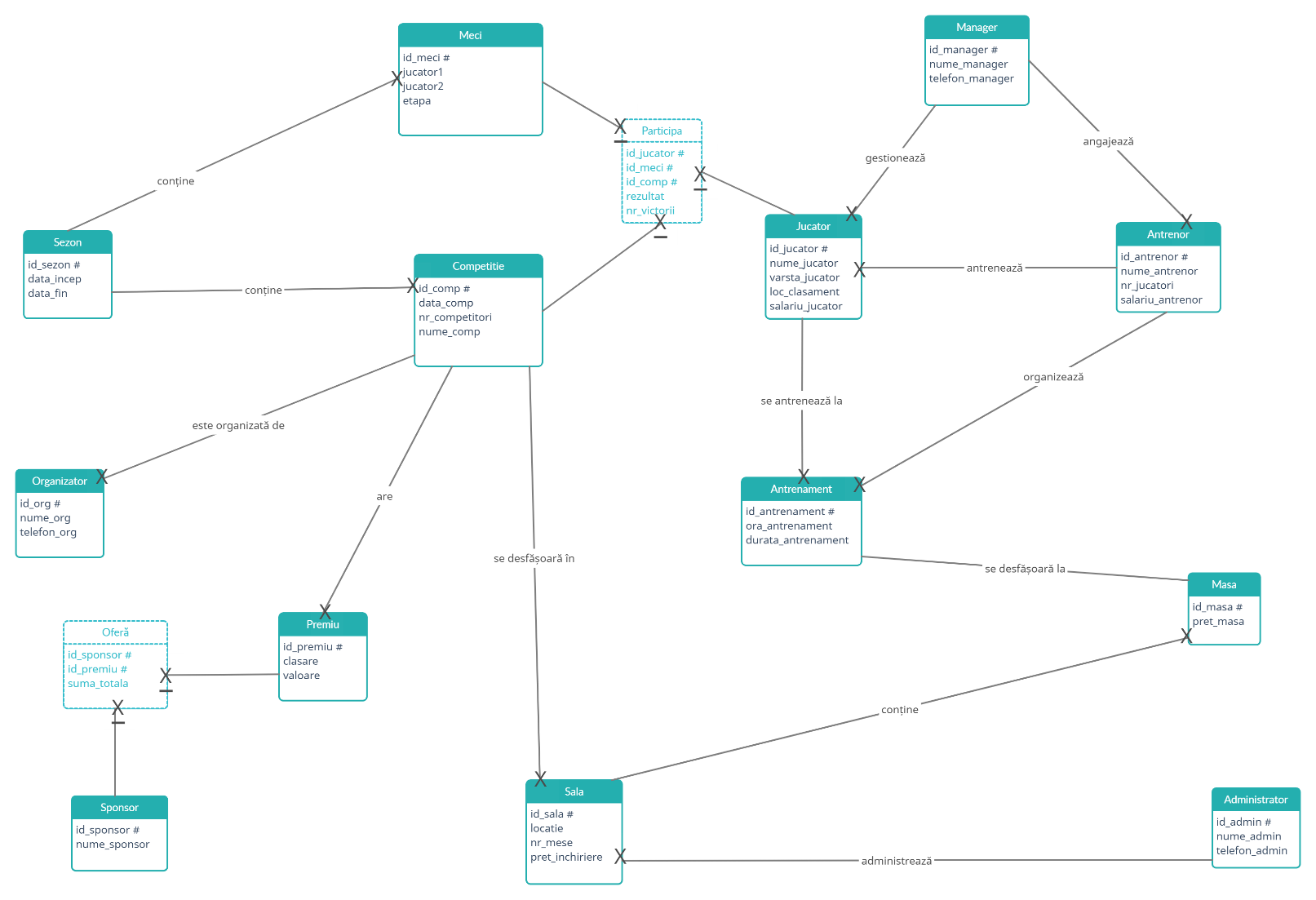
Diagrama Entitate-Relație:

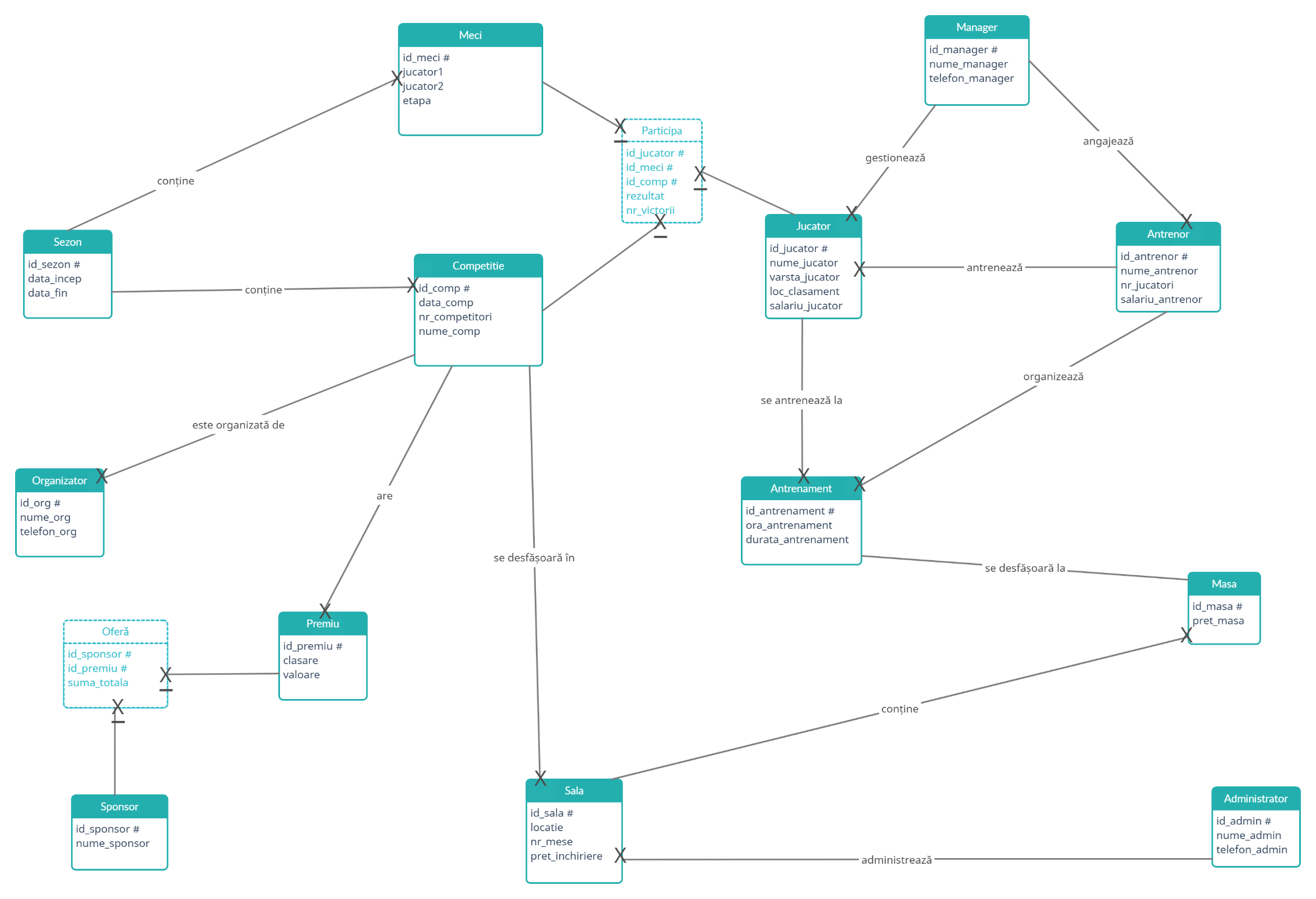




7.

Diagrama conceptuală





8.

Schemele relaționale corespunzătoare diagramei conceptuale sunt urmatoarele:

Manager (id\_manager #, nume\_manager, telefon\_manager)

Antrenor (id\_antrenor #, nume\_antrenor, nr\_jucatori, salariu\_antrenor)

Jucator (id\_jucator #, nume\_jucator, varsta\_jucator, loc\_clasament, salariu\_jucator)

Antrenament (id\_antrenament #, ora antrenament, durata\_antrenament)

Masa (id\_masa #, pret\_masa)

Sala (id\_sala #, locatie, nr\_mese, pret\_inchiriere)

Administrator (id\_admin #, nume\_admin, telefon\_admin)

Competitie (id\_comp #, data\_comp, nr\_competitori, nume\_comp)

Meci (id\_meci #, jucator1, jucator2, etapa)

Participa (id\_jucator #, id\_meci #, id\_comp #, rezultat, nr\_victorii)

Sezon (id\_sezon #, data\_incep, data\_fin)

Organizator (id\_org #, nume\_org, telefon\_org)

Premiu (id\_premiu #, clasare, valoare)

Sponsor(id\_sponsor #, nume\_sponsor)

Ofera (id\_sponsor #, id\_premiu #, suma\_totala)

9.

Realizarea normalizării până la forma normală 3:

Forma normala 1:

O relatie se afla in forma normala 1 (fn1) daca fiecarui atribut care o compune ii corespunde o valoare indivizibila.

Astfel forma normala 1 asigura faptul ca fiecare relatie va avea un identificator unic.

In cadrul clubului de biliard, jucatorii au oportunitatea de a participa la competitii.

Astfel, vom lua drept exemplu entitatile Competitie si Organizator.

|  |  |
| --- | --- |
| Competitie | Organizator |
| Campionatul mondial | Ionescu, Avram |
| Turneul Angliei | Firma1, Firma2, Firma3 |
| Liga Germaniei | Firma2 |

In cele mai multe situatii vom avea mai mult de un organizator pe competitie, astfel valorile ar fi greu de implementat (ar trebui sa parcurgem un sir de caractere din Organizator, ca apoi sa identificam numele fiecarui organizator).

|  |  |
| --- | --- |
| Competitie | Organizator |
| Campionatul mondial | Ionescu |
| Campionatul mondial | Avram |
| Turneul Angliei | Firma1 |
| Turneul Angliei | Firma2 |
| Turneul Angliei | Firma3 |
| Liga Germaniei | Firma2 |

|  |  |  |
| --- | --- | --- |
| Id | Competitie | Organizator |
| 1 | Campionatul mondial | Ionescu |
| 2 | Campionatul mondial | Avram |
| 3 | Turneul Angliei | Firma1 |
| 4 | Turneul Angliei | Firma2 |
| 5 | Turneul Angliei | Firma3 |
| 6 | Liga Germaniei | Firma2 |

Forma normala 2:

O relatie se afla in forma normala 2 (fn2) daca si numai daca aceasta se afla in fn1 si fiecare atribut care nu face parte din cheie este dependent de intreaga cheie primara.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id\_premiu# | id\_sponsor# | clasare | valoare | nume\_sponsor |
| 1 | 1 | 1 | 20000 | Coca-Cola |
| 2 | 2 | 2 | 10000 | Heineken |
| 3 | 2 | 3 | 5000 | Heineken |
| 4 | 2 | 4 | 2000 | Heineken |
| 5 | 3 | 5 | 500 | Lidl |

Relatia este in fn1, deoarece exista identificator unic pentru toate intrarile in tabel.

Pentru a fi in fn2 atributele clasare, valoare, nume\_sponsor trebuie sa depinda de intreaga cheie primara, ceea ce nu se intampla.

Dependentele sunt:

{id\_premiu#} -> {clasare, valoare}

{id\_sponsor#} -> {nume\_sponsor}

Transformare in fn2 => 2 tabele

|  |  |  |
| --- | --- | --- |
| id\_premiu# | clasare | valoare |
| 1 | 1 | 20000 |
| 2 | 2 | 10000 |
| 3 | 3 | 5000 |
| 4 | 4 | 2000 |
| 5 | 5 | 500 |

(Relatia premiu)

|  |  |
| --- | --- |
| id\_sponsor# | nume\_sponsor |
| 1 | Coca-Cola |
| 2 | Heineken |
| 2 | Heineken |
| 2 | Heineken |
| 3 | Lidl |

Luam drept exemplu relatia Premiu (penultimul tabel)

Aceasta relatie se afla in fn1, deoarece exista un identificator unic pentru toate intrarile din tabel.

Se poate remarca faptul ca atributele clasare si valoare nu sunt componente ale cheii primare, dar sunt dependente de aceasta.

Forma normala 3:

O relatie se afla in forma normala 3 (fn3) daca si numai daca se afla in forma normala 2 si fiecare atribut, care nu este cheie, depinde direct de cheia primara si numai de ea.

Vom lua un exemplu imaginar pentru relatia Jucator, unde presupunem ca salariul jucatorului (salariu\_jucator) depinde de locul in clasament (loc\_clasament) al acestuia. (Relatie jucator non-fn3)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| id\_jucator# | nume\_jucator | varsta\_jucator | loc\_clasament | salariu\_jucator |
| 1 | Gheorghe | 30 | 1 | 100000 |
| 2 | Petrescu | 21 | 3 | 20000 |
| 3 | Stoica | 43 | 2 | 60000 |

Aceasta relatie se afla in fn1, deoarece exista un identificator unic pentru toate intrarile din tabel (id\_jucator#).

In plus, relatia se afla si in fn2, fiindca atributele care fac parte din cheia primara sunt depandente de intreaga cheie id\_jucator#.

Putem observa faptul ca atributul salariu\_jucator depinde tranzitiv de cheia primara id\_antrenor# prin atributul loc\_clasament. Astfel relatia nu este in fn3.

Relatiile de dependenta:

{id\_jucator#} -> {nume\_jucator, varsta\_jucator, loc\_clasament, salariu\_jucator}

{id\_jucator#} -> {loc\_clasament} -> {salariu\_jucator}

Tranformare in fn3:

Jucator\_fn3:

|  |  |  |  |
| --- | --- | --- | --- |
| id\_jucator# | nume\_jucator | varsta\_jucator | loc\_clasament |
| 1 | Gheorghe | 30 | 1 |
| 2 | Petrescu | 21 | 3 |
| 3 | Stoica | 43 | 2 |

Loc\_salariu:

|  |  |
| --- | --- |
| loc\_clasament | salariu\_jucator |
| 1 | 100000 |
| 3 | 20000 |
| 2 | 60000 |

10.

Crearea unei secvențe ce va fi utilizată în inserarea înregistrărilor în tabele.

create sequence seq\_id

start with 10

increment by 1

nocycle

nocache;

create table manager (id\_manager number(6) primary key,

nume\_manager varchar2(100) not null,

telefon\_manager varchar(10)

);

insert into manager (id\_manager, nume\_manager, telefon\_manager)

values (seq\_id.nextval, 'Popescu', '0722531164');

insert into manager (id\_manager, nume\_manager, telefon\_manager)

values (seq\_id.nextval, 'Petrescu', '0722931164');

insert into manager (id\_manager, nume\_manager, telefon\_manager)

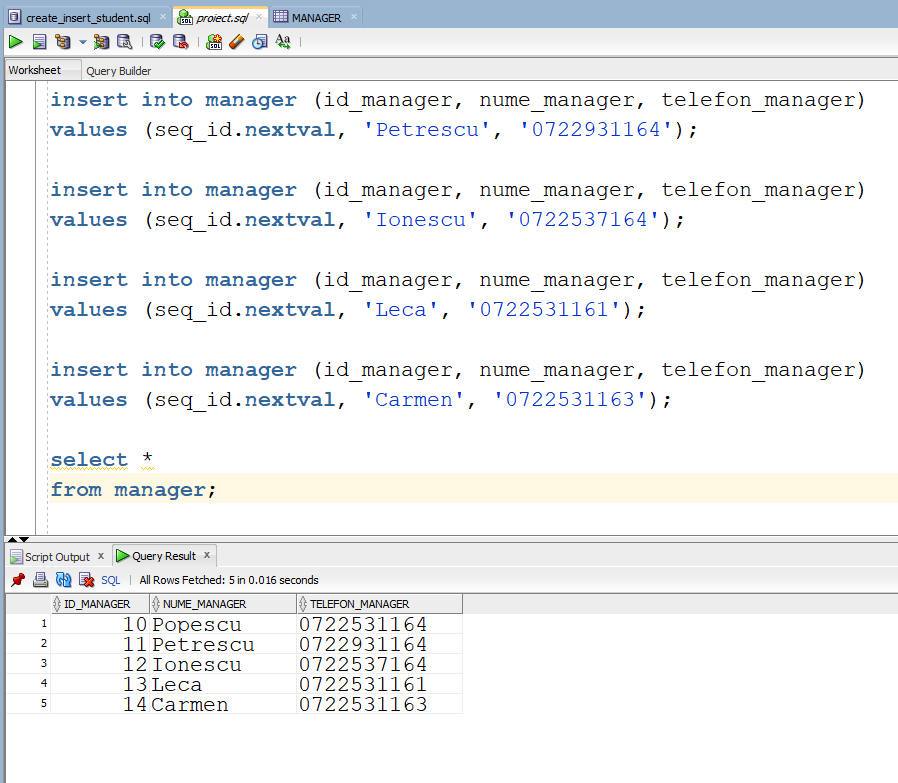
values (seq\_id.nextval, 'Ionescu', '0722537164');

insert into manager (id\_manager, nume\_manager, telefon\_manager)

values (seq\_id.nextval, 'Leca', '0722531161');

insert into manager (id\_manager, nume\_manager, telefon\_manager)

values (seq\_id.nextval, 'Carmen', '0722531163');



11.

Crearea tabelelor în SQL și inserarea de date coerente în fiecare dintre acestea.

(Tabelul Manager a fost prezentat anterior, la exercitiul 10)

create table antrenor (id\_antrenor number(6) primary key,

nume\_antrenor varchar2(100) not null,

nr\_jucatori number(3) not null,

salariu\_antrenor number(10)

);

insert into antrenor (id\_antrenor, nume\_antrenor, nr\_jucatori, salariu\_antrenor)

values (20, 'Sabau', 3, 5000);

insert into antrenor (id\_antrenor, nume\_antrenor, nr\_jucatori, salariu\_antrenor)

values (21, 'Levi', 5, 4500);

insert into antrenor (id\_antrenor, nume\_antrenor, nr\_jucatori, salariu\_antrenor)

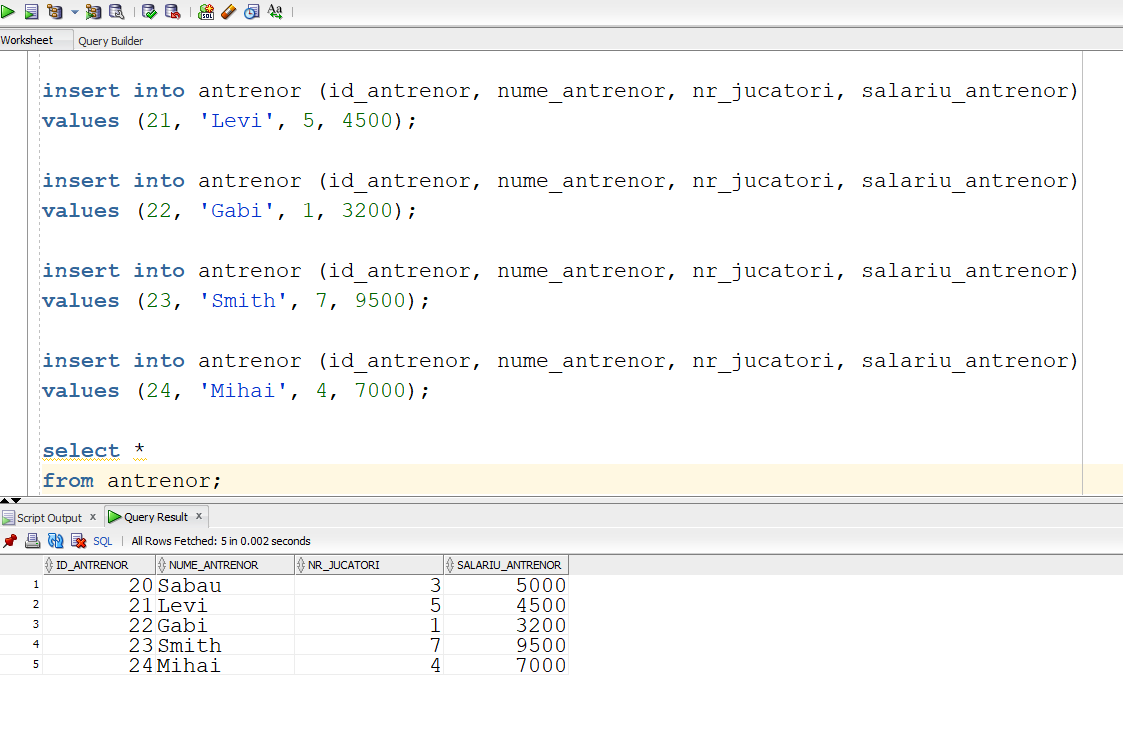
values (22, 'Gabi', 1, 3200);

insert into antrenor (id\_antrenor, nume\_antrenor, nr\_jucatori, salariu\_antrenor)

values (23, 'Smith', 7, 9500);

insert into antrenor (id\_antrenor, nume\_antrenor, nr\_jucatori, salariu\_antrenor)

values (24, 'Mihai', 4, 7000);



create table jucator(id\_jucator number(6) primary key,

nume\_jucator varchar2(100) not null,

varsta\_jucator number(3) not null,

loc\_clasament number(10) not null,

salariu\_jucator number(10)

);

insert into jucator (id\_jucator, nume\_jucator, varsta\_jucator, loc\_clasament, salariu\_jucator)

values (30, 'Ioneci', 30, 5, 2000);

insert into jucator (id\_jucator, nume\_jucator, varsta\_jucator, loc\_clasament, salariu\_jucator)

values (31, 'Danescu', 51, 3, 6000);

insert into jucator (id\_jucator, nume\_jucator, varsta\_jucator, loc\_clasament, salariu\_jucator)

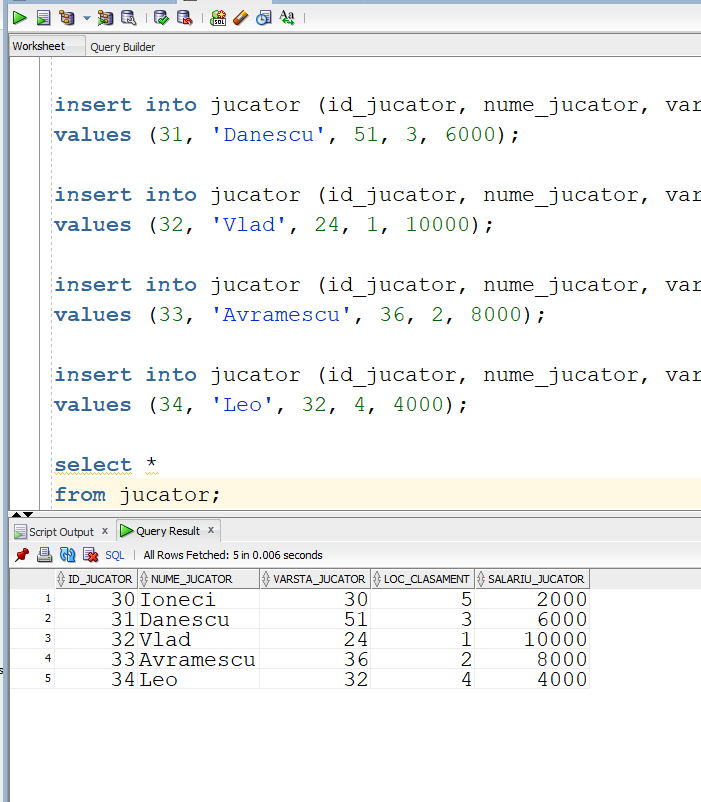
values (32, 'Vlad', 24, 1, 10000);

insert into jucator (id\_jucator, nume\_jucator, varsta\_jucator, loc\_clasament, salariu\_jucator)

values (33, 'Avramescu', 36, 2, 8000);

insert into jucator (id\_jucator, nume\_jucator, varsta\_jucator, loc\_clasament, salariu\_jucator)

values (34, 'Leo', 32, 4, 4000);



create table antrenament(id\_antrenament number(6) primary key,

ora\_antrenament number(2) not null,

durata\_antrenament number(3) not null

);

insert into antrenament (id\_antrenament, ora\_antrenament, durata\_antrenament)

values (40, 14, 180);

insert into antrenament (id\_antrenament, ora\_antrenament, durata\_antrenament)

values (41, 11, 90);

insert into antrenament (id\_antrenament, ora\_antrenament, durata\_antrenament)

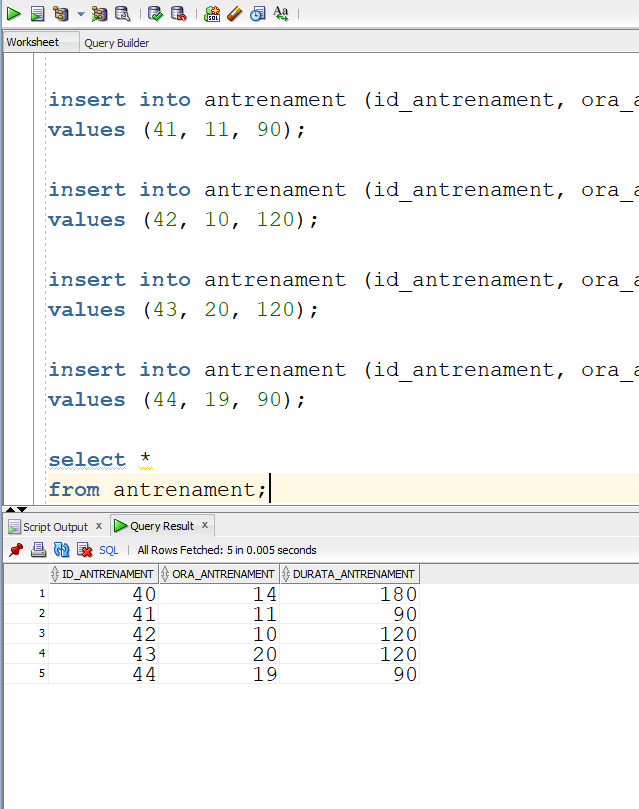
values (42, 10, 120);

insert into antrenament (id\_antrenament, ora\_antrenament, durata\_antrenament)

values (43, 20, 120);

insert into antrenament (id\_antrenament, ora\_antrenament, durata\_antrenament)

values (44, 19, 90);



create table masa(id\_masa number(6) primary key,

pret\_masa number(10)

);

insert into masa(id\_masa, pret\_masa)

values (50, 20000);

insert into masa(id\_masa, pret\_masa)

values (51, 10000);

insert into masa(id\_masa, pret\_masa)

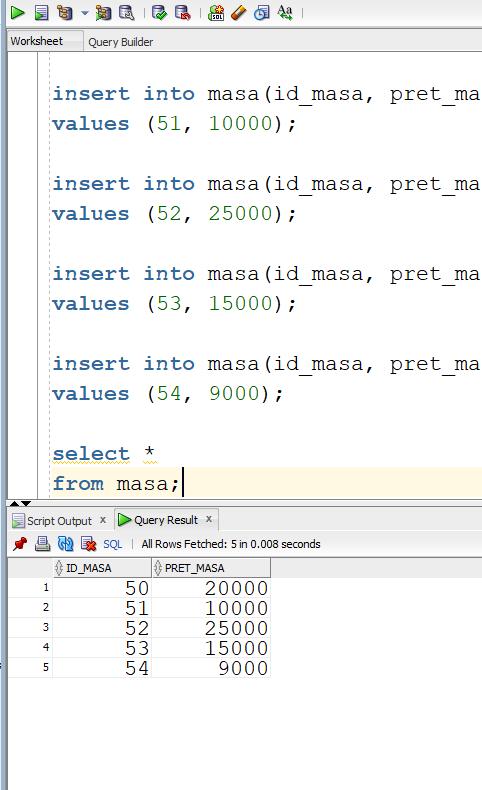
values (52, 25000);

insert into masa(id\_masa, pret\_masa)

values (53, 15000);

insert into masa(id\_masa, pret\_masa)

values (54, 9000);



create table sala(id\_sala number(6) primary key,

locatie varchar2(100) not null,

nr\_mese number(2) not null,

pret\_inchiriere number(10)

);

insert into sala (id\_sala, locatie, nr\_mese, pret\_inchiriere)

values (60, 'Blanari 67', 10, 5000);

insert into sala (id\_sala, locatie, nr\_mese, pret\_inchiriere)

values (61, 'Magheru 90', 20, 10000);

insert into sala (id\_sala, locatie, nr\_mese, pret\_inchiriere)

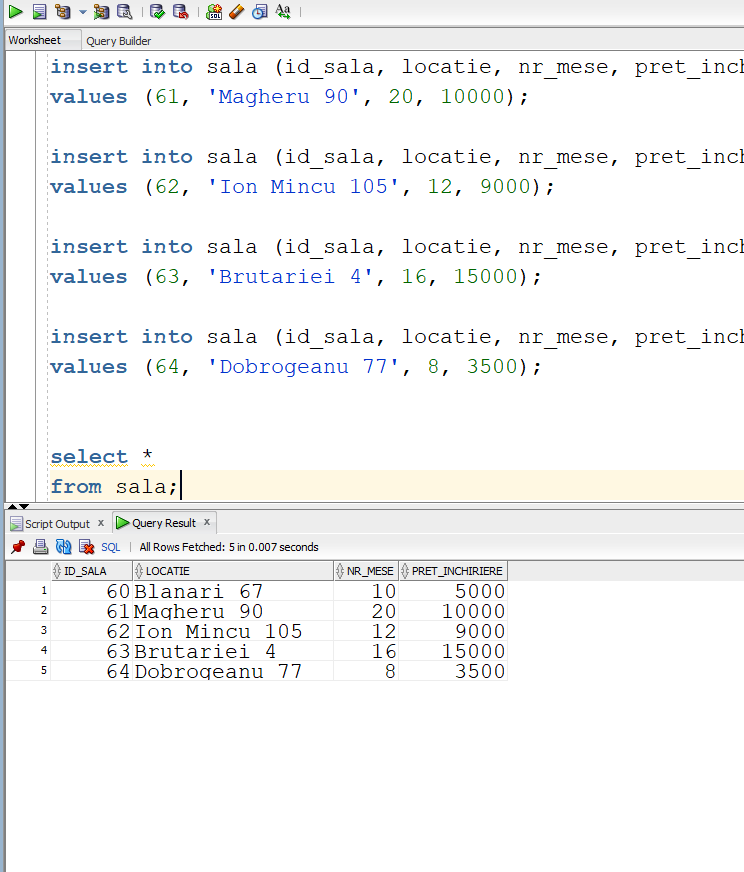
values (62, 'Ion Mincu 105', 12, 9000);

insert into sala (id\_sala, locatie, nr\_mese, pret\_inchiriere)

values (63, 'Brutariei 4', 16, 15000);

insert into sala (id\_sala, locatie, nr\_mese, pret\_inchiriere)

values (64, 'Dobrogeanu 77', 8, 3500);



create table administrator(id\_admin number(6) primary key,

nume\_admin varchar2(100) not null,

telefon\_admin number(10) );

insert into administrator(id\_admin, nume\_admin, telefon\_admin)

values (70, 'Petre', 0735899641);

insert into administrator(id\_admin, nume\_admin, telefon\_admin)

values (71, 'Pavel', 0735822641);

insert into administrator(id\_admin, nume\_admin, telefon\_admin)

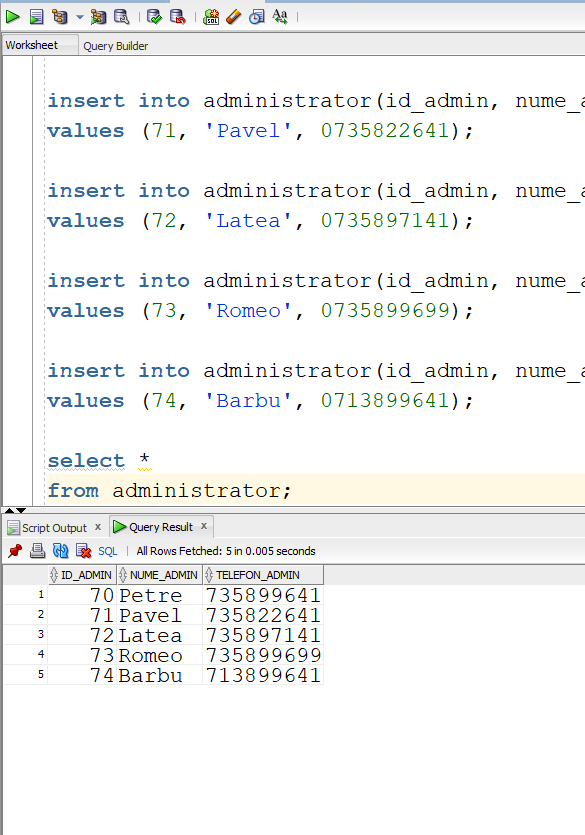
values (72, 'Latea', 0735897141);

insert into administrator(id\_admin, nume\_admin, telefon\_admin)

values (73, 'Romeo', 0735899699);

insert into administrator(id\_admin, nume\_admin, telefon\_admin)

values (74, 'Barbu', 0713899641);



create table meci(id\_meci number(6) primary key,

jucator1 varchar2(100) not null,

jucator2 varchar2(100) not null,

nr\_jocuri number(2)

);

insert into meci(id\_meci, jucator1, jucator2, nr\_jocuri)

values (80, 'Alex', 'Petru', 9);

insert into meci(id\_meci, jucator1, jucator2, nr\_jocuri)

values (81, 'Andrei', 'Liviu', 11);

insert into meci(id\_meci, jucator1, jucator2, nr\_jocuri)

values (82, 'Bogdan', 'Rif', 15);

insert into meci(id\_meci, jucator1, jucator2, nr\_jocuri)

values (83, 'George', 'Constantin', 5);

insert into meci(id\_meci, jucator1, jucator2, nr\_jocuri)

values (84, 'Nicolas', 'Marius', 7);

insert into meci(id\_meci, jucator1, jucator2, nr\_jocuri)

values (200, 'Ioneci', 'Danescu', 9);

insert into meci(id\_meci, jucator1, jucator2, nr\_jocuri)

values (201, 'Ioneci', 'Vlad', 9);

insert into meci(id\_meci, jucator1, jucator2, nr\_jocuri)

values (202, 'Ioneci', 'Avramescu', 9);

insert into meci(id\_meci, jucator1, jucator2, nr\_jocuri)

values (203, 'Ioneci', 'Leo', 9);

insert into meci(id\_meci, jucator1, jucator2, nr\_jocuri)

values (204, 'Danescu', 'Vlad', 9);

insert into meci(id\_meci, jucator1, jucator2, nr\_jocuri)

values (205, 'Danescu', 'Avramescu', 9);

insert into meci(id\_meci, jucator1, jucator2, nr\_jocuri)

values (206, 'Danescu', 'Leo', 9);

insert into meci(id\_meci, jucator1, jucator2, nr\_jocuri)

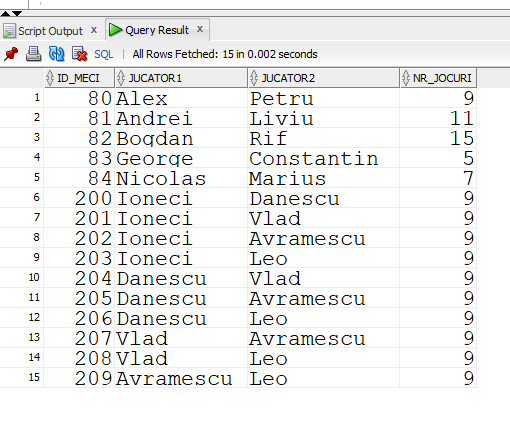
values (207, 'Vlad', 'Avramescu', 9);

insert into meci(id\_meci, jucator1, jucator2, nr\_jocuri)

values (208, 'Vlad', 'Leo', 9);

insert into meci(id\_meci, jucator1, jucator2, nr\_jocuri)

values (209, 'Avramescu', 'Leo', 9);



create table participa(id\_jucator number(6) not null,

id\_meci number(6) not null,

id\_comp number(6) not null,

rezultat varchar2(100) not null,

nr\_victorii number(2)

);

insert into participa (id\_jucator, id\_meci, id\_comp, rezultat, nr\_victorii)

values (30, 200, 90, 'Ioneci', 2);

insert into participa (id\_jucator, id\_meci, id\_comp, rezultat, nr\_victorii)

values (30, 201, 90, 'Vlad', 2);

insert into participa (id\_jucator, id\_meci, id\_comp, rezultat, nr\_victorii)

values (30, 202, 90, 'Ioneci', 2);

insert into participa (id\_jucator, id\_meci, id\_comp, rezultat, nr\_victorii)

values (34, 203, 90, 'Leo', 2);

insert into participa (id\_jucator, id\_meci, id\_comp, rezultat, nr\_victorii)

values (31, 204, 90, 'Danescu', 2);

insert into participa (id\_jucator, id\_meci, id\_comp, rezultat, nr\_victorii)

values (31, 205, 90, 'Avramescu', 2);

insert into participa (id\_jucator, id\_meci, id\_comp, rezultat, nr\_victorii)

values (31, 206, 90, 'Danescu', 2);

insert into participa (id\_jucator, id\_meci, id\_comp, rezultat, nr\_victorii)

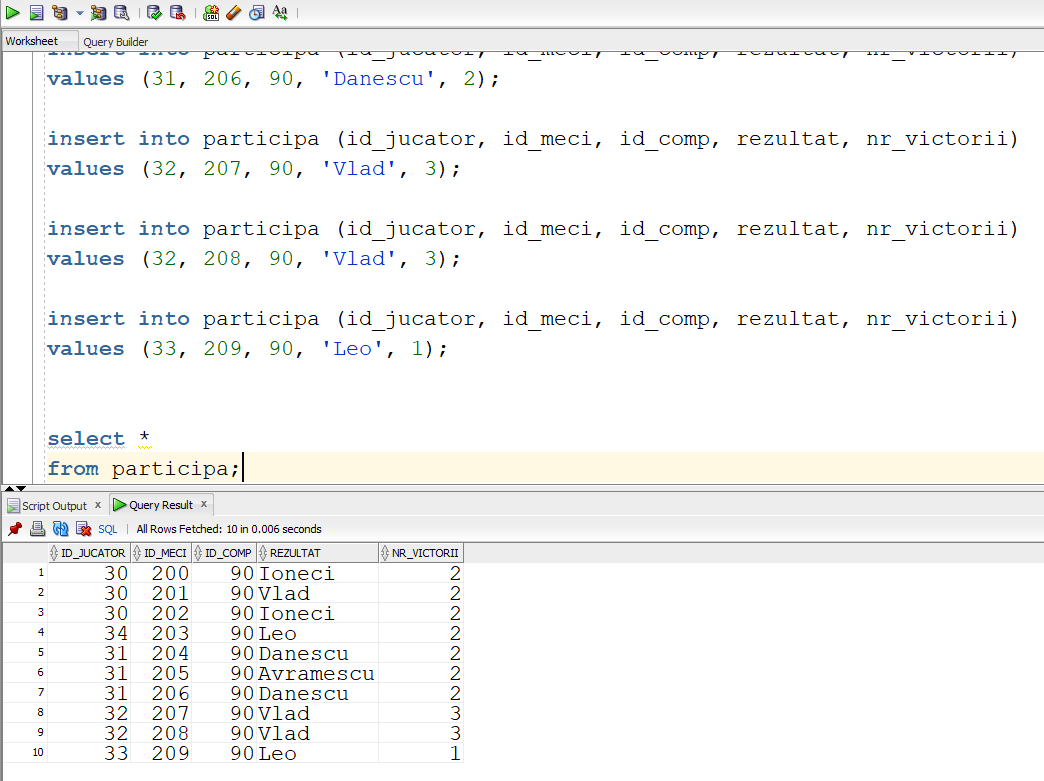
values (32, 207, 90, 'Vlad', 3);

insert into participa (id\_jucator, id\_meci, id\_comp, rezultat, nr\_victorii)

values (32, 208, 90, 'Vlad', 3);

insert into participa (id\_jucator, id\_meci, id\_comp, rezultat, nr\_victorii)

values (33, 209, 90, 'Leo', 1);



create table competitie(id\_comp number(6) primary key,

data\_comp date not null,

nr\_competitori number(5) not null,

nume\_comp varchar2(100) not null

);

insert into competitie(id\_comp, data\_comp, nr\_competitori, nume\_comp)

values (90, to\_date('20-04-2019', 'DD-MM-YYYY'), 128, 'Campionatul Mondial');

insert into competitie(id\_comp, data\_comp, nr\_competitori, nume\_comp)

values (91, to\_date('15-05-2022', 'DD-MM-YYYY'), 16, 'Liga Legendelor');

insert into competitie(id\_comp, data\_comp, nr\_competitori, nume\_comp)

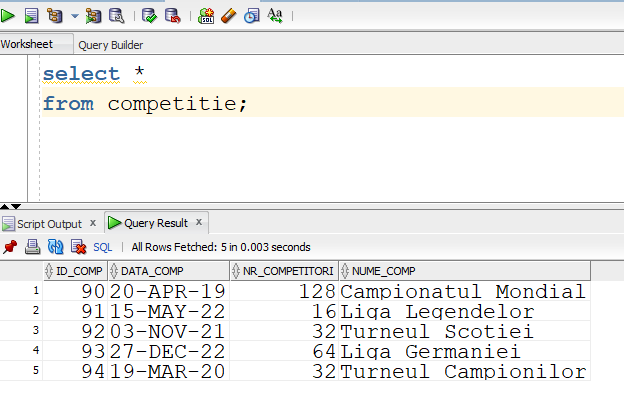
values (92, to\_date('03-11-2021', 'DD-MM-YYYY'), 32, 'Turneul Scotiei');

insert into competitie(id\_comp, data\_comp, nr\_competitori, nume\_comp)

values (93, to\_date('27-12-2022', 'DD-MM-YYYY'), 64, 'Liga Germaniei');

insert into competitie(id\_comp, data\_comp, nr\_competitori, nume\_comp)

values (94, to\_date('19-03-2020', 'DD-MM-YYYY'), 32, 'Turneul Campionilor');



create table sezon(id\_sezon number(6) primary key,

data\_incep date not null,

data\_fin date not null

);

insert into sezon (id\_sezon, data\_incep, data\_fin)

values (100, to\_date('28-09-2018', 'DD-MM-YYYY'), to\_date('27-06-2019', 'DD-MM-YYYY'));

insert into sezon (id\_sezon, data\_incep, data\_fin)

values (101, to\_date('28-09-2019', 'DD-MM-YYYY'), to\_date('27-06-2020', 'DD-MM-YYYY'));

insert into sezon (id\_sezon, data\_incep, data\_fin)

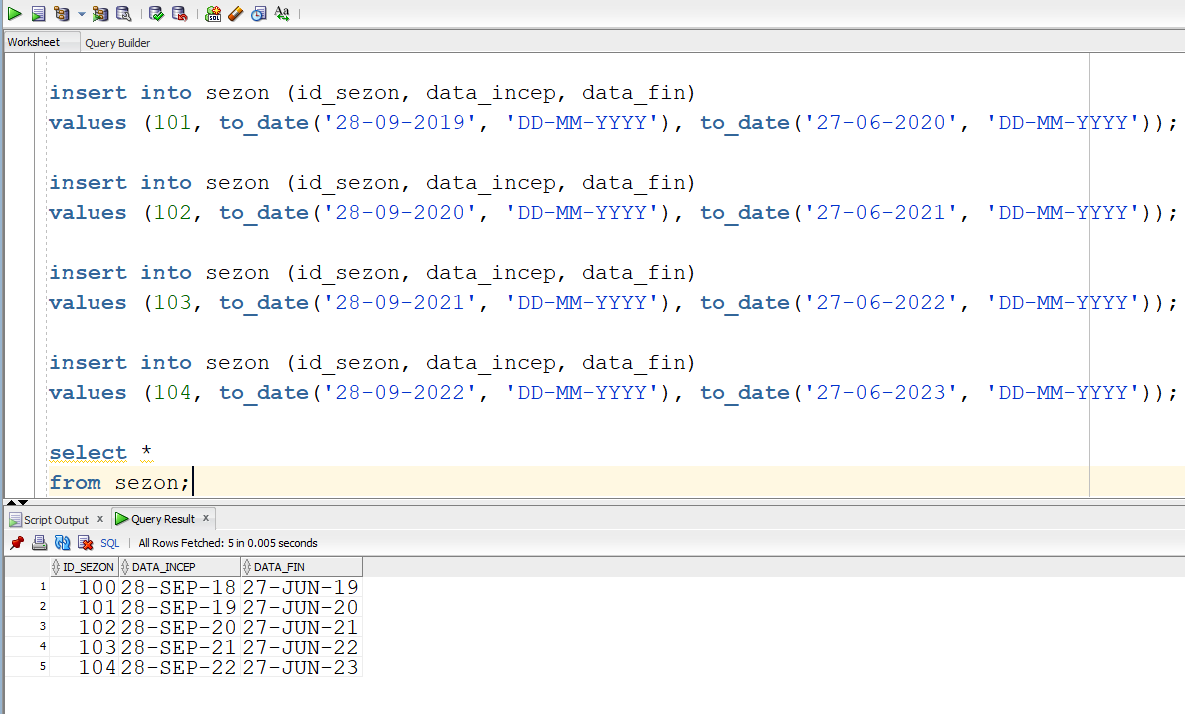
values (102, to\_date('28-09-2020', 'DD-MM-YYYY'), to\_date('27-06-2021', 'DD-MM-YYYY'));

insert into sezon (id\_sezon, data\_incep, data\_fin)

values (103, to\_date('28-09-2021', 'DD-MM-YYYY'), to\_date('27-06-2022', 'DD-MM-YYYY'));

insert into sezon (id\_sezon, data\_incep, data\_fin)

values (104, to\_date('28-09-2022', 'DD-MM-YYYY'), to\_date('27-06-2023', 'DD-MM-YYYY'));



create table organizator (id\_org number(6) primary key,

nume\_org varchar2(100) not null,

telefon\_org varchar(11)

);

insert into organizator (id\_org, nume\_org, telefon\_org)

values (110, 'Popici', '0762531164');

insert into organizator (id\_org, nume\_org, telefon\_org)

values (111, 'Olivia', '0772931164');

insert into organizator (id\_org, nume\_org, telefon\_org)

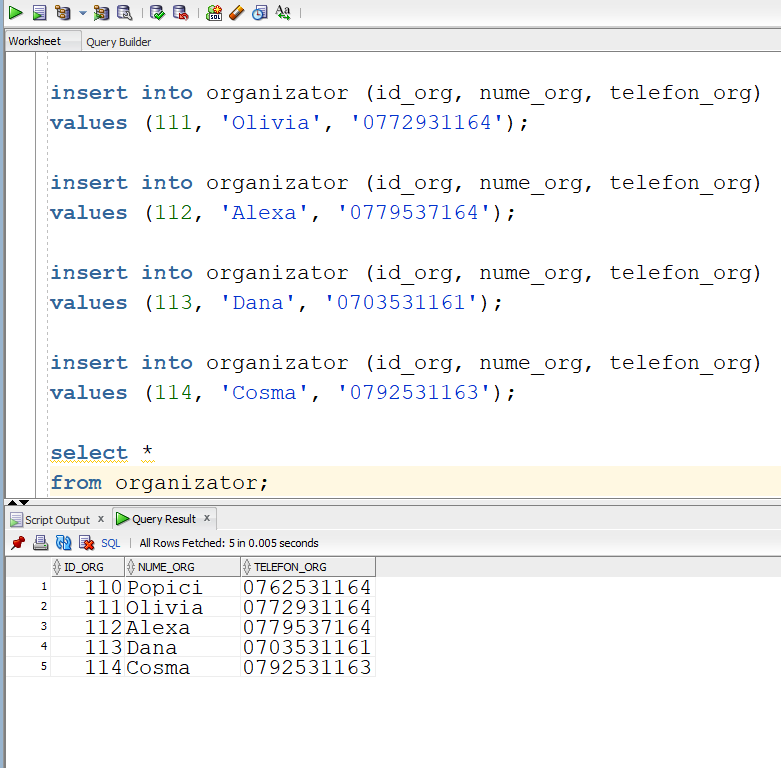
values (112, 'Alexa', '0779537164');

insert into organizator (id\_org, nume\_org, telefon\_org)

values (113, 'Dana', '0703531161');

insert into organizator (id\_org, nume\_org, telefon\_org)

values (114, 'Cosma', '0792531163');



create table premiu (id\_premiu number(6) primary key,

clasare number(3),

valoare number(10) not null

);

insert into premiu (id\_premiu, clasare, valoare)

values (120, 1, 250000);

insert into premiu (id\_premiu, clasare, valoare)

values (121, 2, 100000);

insert into premiu (id\_premiu, clasare, valoare)

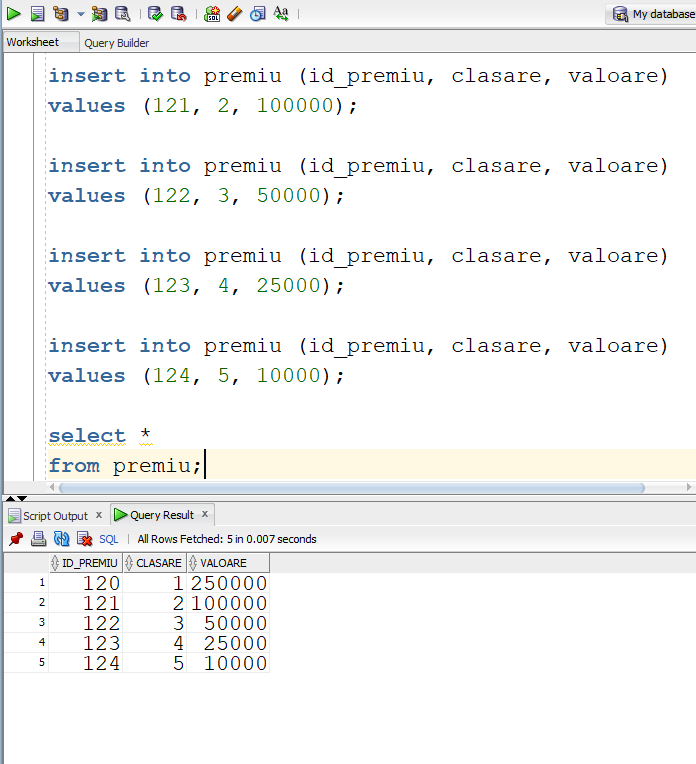
values (122, 3, 50000);

insert into premiu (id\_premiu, clasare, valoare)

values (123, 4, 25000);

insert into premiu (id\_premiu, clasare, valoare)

values (124, 5, 10000);



create table sponsor (id\_sponsor number(6) primary key,

nume\_sponsor varchar2(100) not null

);

insert into sponsor (id\_sponsor, nume\_sponsor)

values (130, 'Coca Cola');

insert into sponsor (id\_sponsor, nume\_sponsor)

values (131, 'Aqua Carpatica');

insert into sponsor (id\_sponsor, nume\_sponsor)

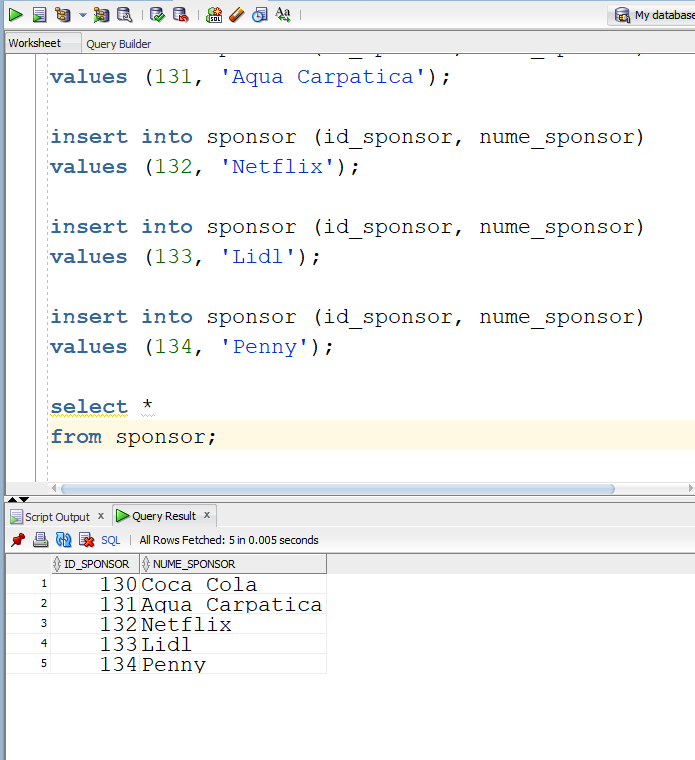
values (132, 'Netflix');

insert into sponsor (id\_sponsor, nume\_sponsor)

values (133, 'Lidl');

insert into sponsor (id\_sponsor, nume\_sponsor)

values (134, 'Penny');



12.

Formulați în limbaj natural și implementați 5 cereri SQL complexe ce vor utiliza, în ansamblul lor, următoarele elemente:

• operație join pe cel puțin 4 tabele

* 1. • filtrare la nivel de linii
  2. • subcereri sincronizate în care intervin cel puțin 3 tabele
  3. • subcereri nesincronizate în care intervin cel puțin 3 tabele
  4. • grupări de date, funcții grup, filtrare la nivel de grupuri
  5. • ordonări
  6. • utilizarea a cel puțin 2 funcții pe șiruri de caractere, 2 funcții pe date calendaristice, a funcțiilor NVL și DECODE, a cel puțin unei expresii CASE
  7. • utilizarea a cel puțin 1 bloc de cerere (clauza WITH)

I.

--Aflati numele, locul in clasament si salariul jucatorului cu cele mai multe victorii

--la 'Campionatul Mondial', premiul castigat de acesta datorita clasarii sale si numele sponsorilor care il ofera.

--Sponsorul cu id-ul cel mai mic ofera premiul 1, sponsorul cu al doilea cel mai mic id ofera premiul 2 si asa mai departe.

select j.nume\_jucator, j.loc\_clasament, j.salariu\_jucator, p.valoare, s.nume\_sponsor

from jucator j, premiu p, sponsor s

where j.id\_jucator in (select id\_jucator

from participa

where nr\_victorii = (select max(nr\_victorii)

from participa))

and

p.clasare = (select loc\_clasament

from jucator

where id\_jucator in (select id\_jucator

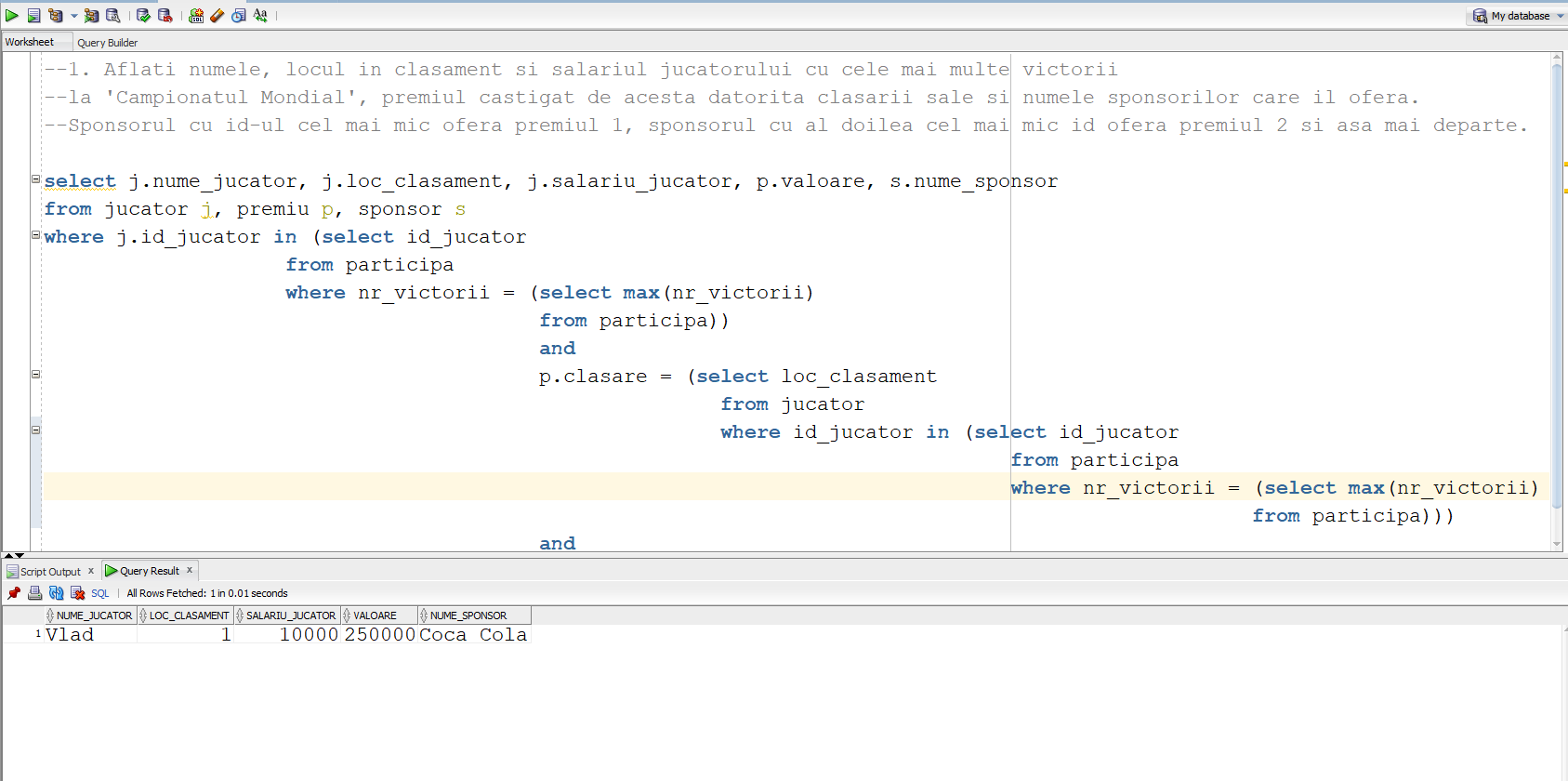
from participa

where nr\_victorii = (select max(nr\_victorii)

from participa)))

and

id\_sponsor = p.clasare + 129;



II.

--Sa se afiseze numarul de competitori si data inceperii competitiei in care jucatorii,

--care au un salariu mai mare decat media salariilor antrenorilor, au castigat macar un meci.

select nr\_competitori, to\_char(data\_comp) "Data inceperii competitiei"

from competitie

where id\_comp in (select id\_comp

from participa

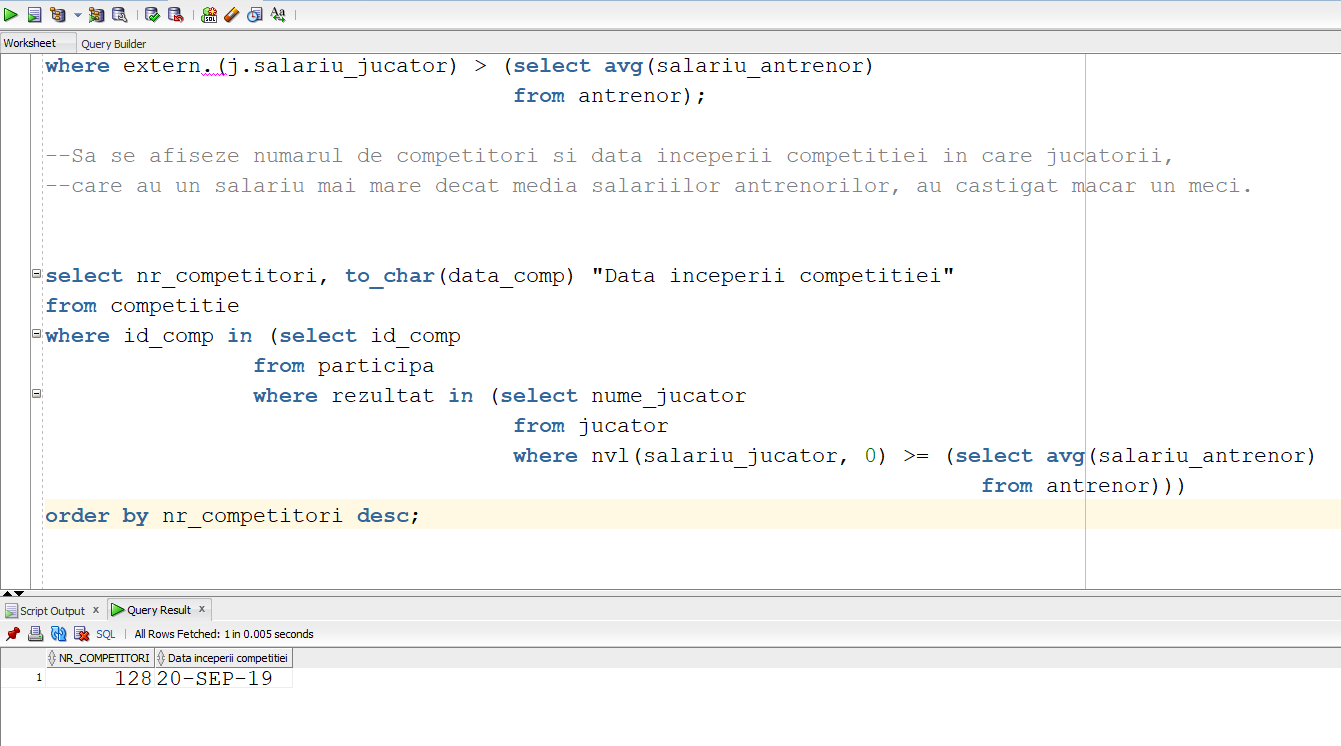
where rezultat in (select nume\_jucator

from jucator

where nvl(salariu\_jucator, 0) >= (select avg(salariu\_antrenor)

from antrenor)))

order by nr\_competitori desc;



III.

--Deoarece la un moment dat bugetul a fost limitat, Ghiocel (organizatorul sezonului de biliard)

--a inchiriat o sala care sa aiba pretul mai mic de 500/masa, pentru competitiile

--cu peste 100 de competitori. Dorim sa aflam id-ul, data de inceup si data de final a sezoanelor

--in care s-au tinut competitiile cu buget limitat.

with f as (select nr\_mese, pret\_inchiriere

from sala)

select id\_sezon, data\_incep, data\_fin

from sezon

where data\_incep < (select data\_comp

from competitie

where nr\_competitori > 100 and 10 > (select nr\_mese

from f

where pret\_inchiriere/nr\_mese < 500))

and

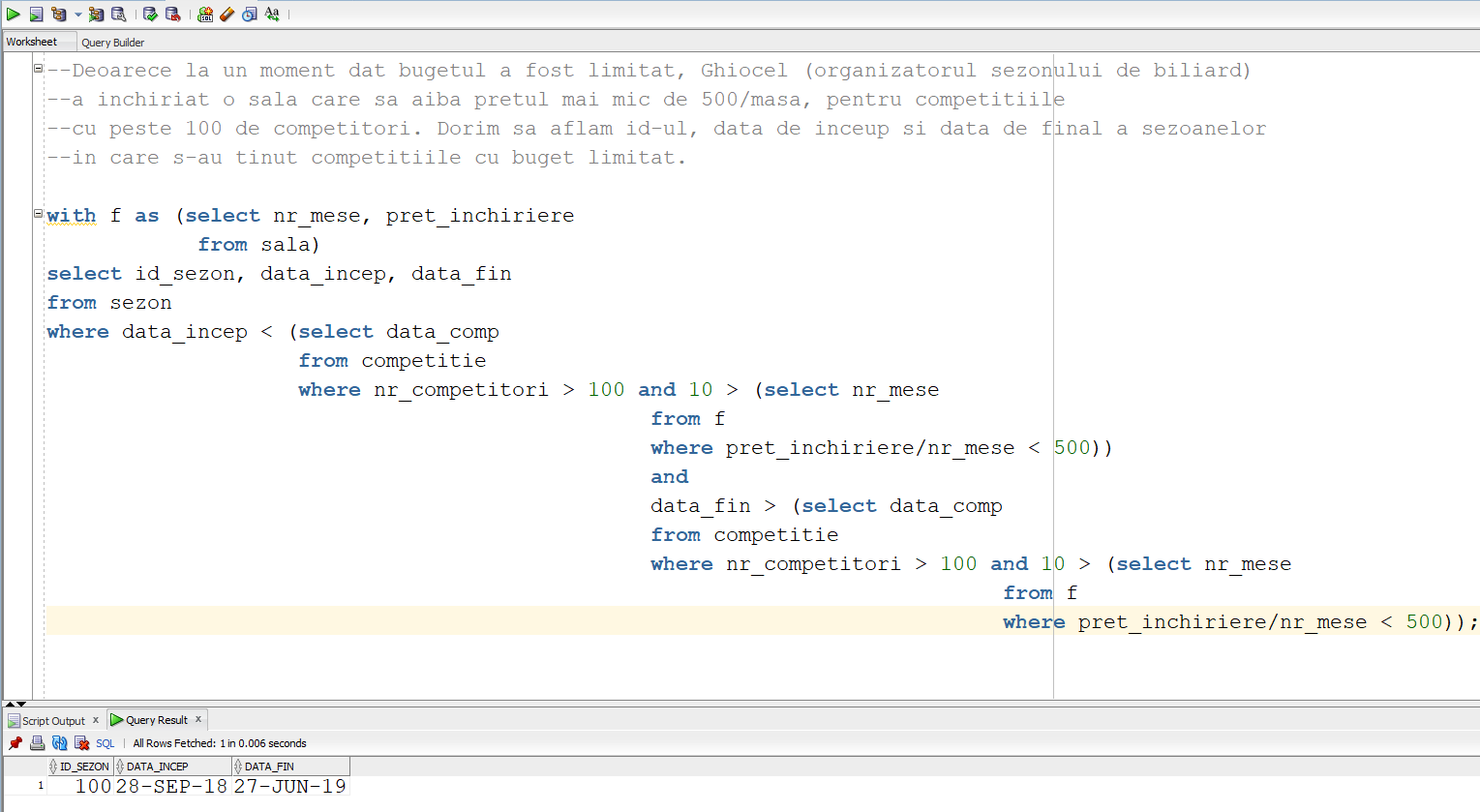
data\_fin > (select data\_comp

from competitie

where nr\_competitori > 100 and 10 > (select nr\_mese

from f

where pret\_inchiriere/nr\_mese < 500));



IV.

--Sa se afle numele, anul nasterii si salariul jucatorului care a avut cea mai slaba performanta la competitii, adica a avut cele mai putine victorii si a jucatorului de pe ultima pozitie in clasamentul mondial.

select nume\_jucator, 2022 - varsta\_jucator "Anul nasterii", salariu\_jucator

from jucator, dual

where id\_jucator = (select id\_jucator

from participa

where nr\_victorii = (select min(nr\_victorii)

from participa))

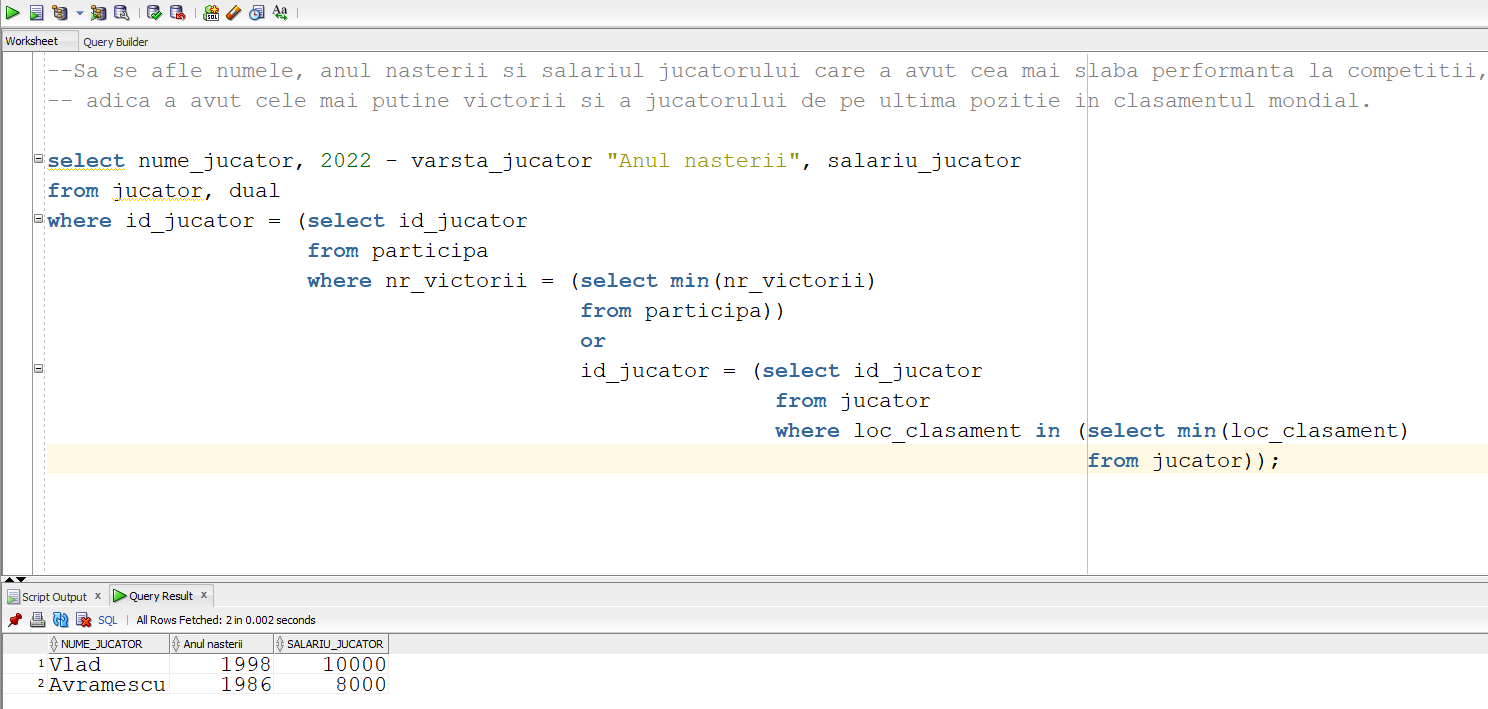
or

id\_jucator = (select id\_jucator

from jucator

where loc\_clasament in (select min(loc\_clasament)

from jucator));



V.

--Afisarea celor 2 jucatori care au participat in meciurile castigate

--de cel mai slab platit jucator de la clubul de biliard

select jucator1, jucator2

from meci

where id\_meci = (select id\_meci

from participa

where rezultat = (select id\_jucator

from jucator

where nvl(salariu\_jucator, 0) = (select min(nvl(salariu\_jucator, 0))

from jucator)));

13.

Implementarea a 3 operații de actualizare și de suprimare a datelor utilizând subcereri:

I.

--Din pacate au fost facute micsorari de buget la

--clubul nostru de biliard, astfel nu ne mai putem

--permite sa platim jucator aflat pe ultima pozitie

--in clasament.

update jucator

set salariu\_jucator = null

where loc\_clasament = (select min(loc\_clasament)

from jucator);

rollback;

II.

--Conform regulamentului, jucatorii care nu au castigat

--niciun meci de biliard isi vor pierde licenta de

--profesionisti, precum si locul din clubul de biliard.

delete jucator

where id\_jucator not in (select rezultat

from participa);

rollback;

III.

--Administratorul salilor de biliard a observat ca

--salile de marime medie (cu 8-11 mese) se aglomereaza

--destul de mult in cursul dupa-amiezii, asa ca s-a

--gandit ca este nevoie de o extindere.

update sala

set nr\_mese = 15

where nr\_mese < 12 and nr\_mese > 7;

rollback;