**Campionat Mondial de Fotbal 2014**

**Cerinta 1**

Baza de date a Campionatului Mondial de Fotbal din 2014 contine toate cele 32 de echipe calificate in faza grupelor si sposnorii acestora. Pentru fiecare echipa calificata mai departe din faza grupelor(primele 2 echipe) avem datele despre antrenorii acestora. Meciurile din fazele eliminatorii dispun de bilete caracteristice fazei competitiei. Acestea sunt de tipul standard si VIP. Fazele eliminatorii ale Campionatului Mondial se vor disputa pe parcursul lunii iulie . Un meci este televizat de mai multe canalte tv, in functie de importanta acestuia. Meciurile terminate la egalitate se vor decide la lovituri de departajare.

De asemena baza de date prezinta date despre cei mai buni 10 jucatori de la campionatul Mondial.

Cerinta 2

Diagram, schematic

Description automatically generated

**Diagram

Description automatically generated**Cerinta 3

Cerinta 4

create table tari(

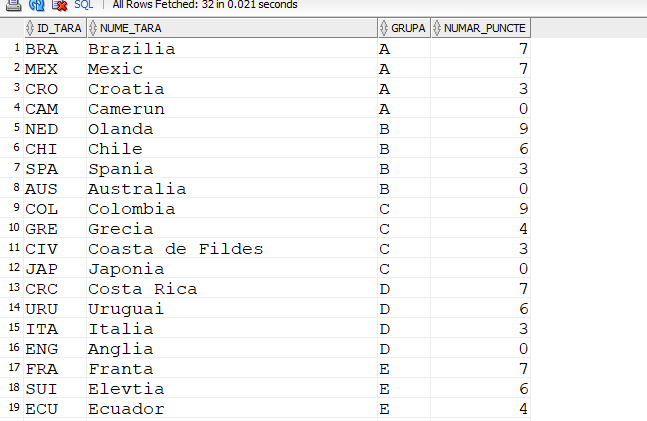
id\_tara varchar2(5) not null primary KEY,

nume\_tara varchar2(50),

grupa char(1),

numar\_puncte int

);



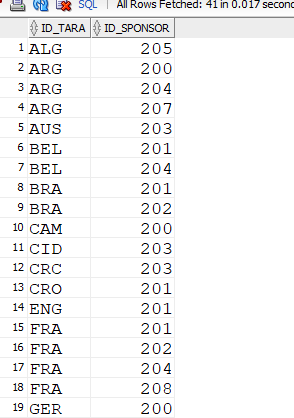
create table sponsorizeaza(

id\_tara varchar2(5) not null,

id\_sponsor int not null,

constraint pk\_sponsorizeaza primary key(id\_tara,id\_sponsor)

);

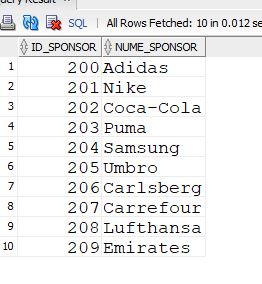


create table sponsori(

id\_sponsor int not null primary KEY,

nume\_sponsor varchar(30)

);



create table meci(

id\_meci int not null primary key,

etapa varchar2(20),

data\_meciului date not null,

id\_stadion int not null,

id\_echipa1 varchar2(50),

id\_echipa2 varchar2(50),

goluri1 int,

goluri2 int,

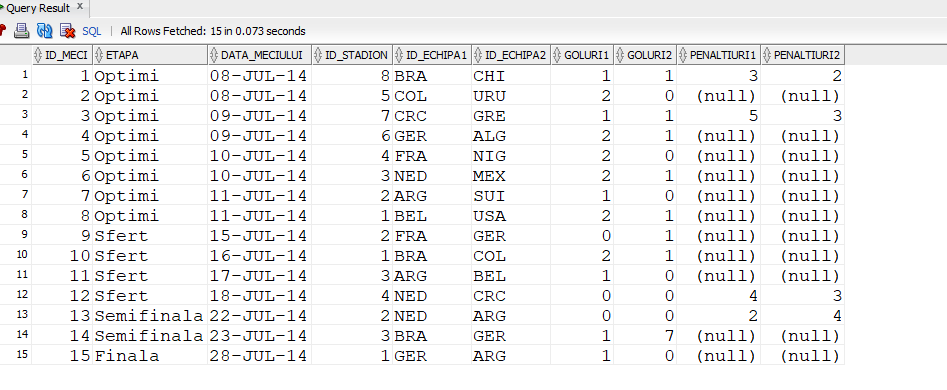
penaltiuri1 int,

penaltiuri2 int

);

alter table meci add foreign key(id\_echipa1) references tari(id\_tara);

alter table meci add foreign key(id\_echipa2) references tari(id\_tara);



create table stadion(

id\_stadion int not null primary key,

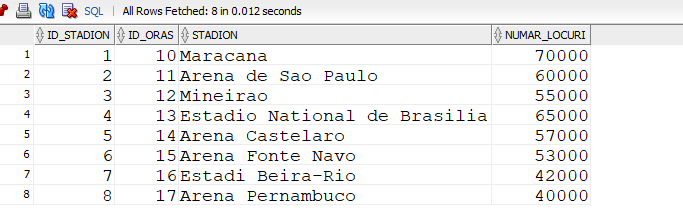
id\_oras int,

stadion varchar2(50),

numar\_locuri int

);

alter table stadion add foreign key(id\_oras) REFERENCES orase(id\_oras);



create table Orase(

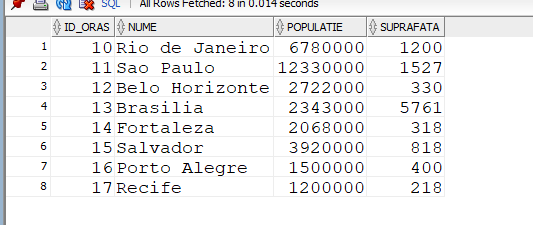
id\_oras int not null primary key,

nume varchar2(50),

populatie int,

suprafata float

);

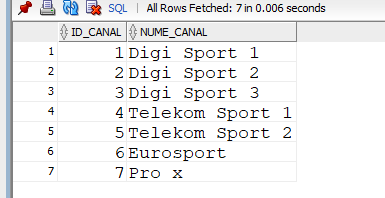


create table canale\_tv(

id\_canal int not null primary key,

nume\_canal varchar2(30)

);



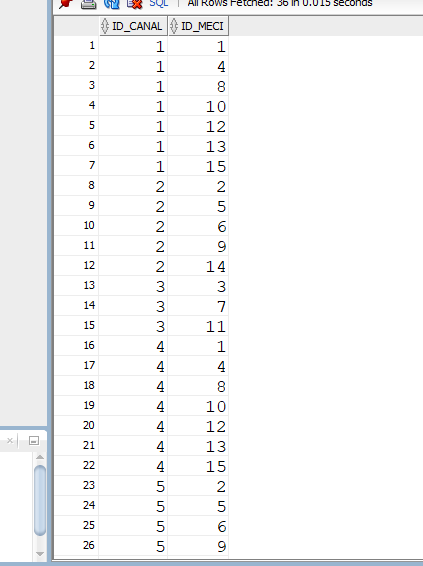
create table transmite(

id\_canal int not null,

id\_meci int not null,

constraint pk\_transmite primary key(id\_canal,id\_meci)

);



create table jucatori(

id\_jucator int not null primary key,

nume varchar2(20),

prenume varchar2(20),

goluri int,

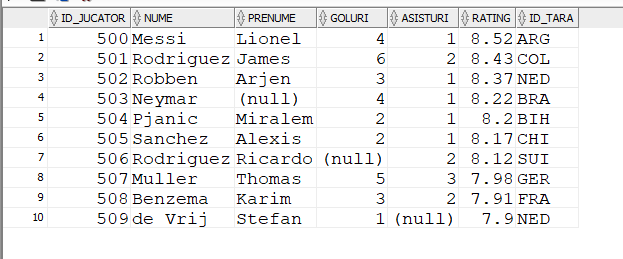
asisturi int,

rating float,

id\_tara varchar2(50),

foreign key(id\_tara) references tari(id\_tara)

);



create table antrenori(

id\_antrenor int not null primary key,

nume varchar2(20),

prenume varchar2(20),

id\_tara varchar2(50),

foreign key(id\_tara) references tari(id\_tara)

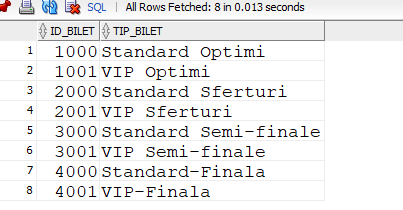
);



create table bilete(

id\_bilet int not null primary key,

tip\_bilet varchar2(50));

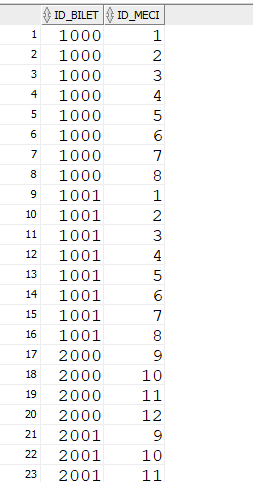


create table dispune(

id\_bilet int,

id\_meci int,

constraint pk\_dispune primary key(id\_bilet,id\_meci));



Cerinta 5

--tari

insert into tari

values('BRA','Brazilia','A',7);

insert into tari

values('MEX','Mexic','A',7);

insert into tari

values('CRO','Croatia','A',3);

insert into tari

values('CAM','Camerun','A',0);

insert into tari

values('NED','Olanda','B',9);

insert into tari

values('CHI','Chile','B',6);

insert into tari

values('SPA','Spania','B',3);

insert into tari

values('AUS','Australia','B',0);

insert into tari

values('COL','Colombia','C',9);

insert into tari

values('GRE','Grecia','C',4);

insert into tari

values('CIV','Coasta de Fildes','C',3);

insert into tari

values('JAP','Japonia','C',0);

insert into tari

values('CRC','Costa Rica','D',7);

insert into tari

values('URU','Uruguai','D',6);

insert into tari

values('ITA','Italia','D',3);

insert into tari

values('ENG','Anglia','D',0);

insert into tari

values('FRA','Franta','E',7);

insert into tari

values('SUI','Elevtia','E',6);

insert into tari

values('ECU','Ecuador','E',4);

insert into tari

values('HON','Honduras','E',0);

insert into tari

values('ARG','Argentina','F',9);

insert into tari

values('NIG','Nigeria','F',4);

insert into tari

values('BIH','Bosnia si Hertegovina','F',3);

insert into tari

values('IRA','Iran','F',1);

insert into tari

values('GER','Germania','G',7);

insert into tari

values('USA','Statele Unite ale Americii','G',5);

insert into tari

values('POR','Portugal','G',4);

insert into tari

values('GHA','Ghana','G',1);

insert into tari

values('BEL','Belgium','H',9);

insert into tari

values('ALG','Algeria','H',4);

insert into tari

values('RUS','Rusia','H',2);

insert into tari

values('KOR','Coreea de Sud','H',1);

--sponsori

insert into sponsori

values(200,'Adidas');

insert into sponsori

values(201,'Nike');

insert into sponsori

values(202,'Coca-Cola');

insert into sponsori

values(203,'Puma');

insert into sponsori

values(204,'Samsung');

insert into sponsori

values(205,'Umbro');

insert into sponsori

values(206,'Carlsberg');

insert into sponsori

values(207,'Carrefour');

insert into sponsori

values(208,'Lufthansa');

insert into sponsori

values(209,'Emirates');

--sponsorizeaza

--jucatori

insert into jucatori

values(500,'Messi','Lionel',4,1,8.52,'ARG');

insert into jucatori

values(501,'Rodriguez','James',6,2,8.43,'COL');

insert into jucatori

values(502,'Robben','Arjen',3,1,8.37,'NED');

insert into jucatori

values(503,'Neymar',null,4,1,8.22,'BRA');

insert into jucatori

values(504,'Pjanic','Miralem',2,1,8.2,'BIH');

insert into jucatori

values(505,'Sanchez','Alexis',2,1,8.17,'CHI');

insert into jucatori

values(506,'Rodriguez','Ricardo',null,2,8.12,'SUI');

insert into jucatori

values(507,'Muller','Thomas',5,3,7.98,'GER');

insert into jucatori

values(508,'Benzema','Karim',3,2,7.91,'FRA');

insert into jucatori

values(509,'de Vrij','Stefan',1,null,7.9,'NED');

--stadioane

create SEQUENCE nr

start with 1

increment by 1;

insert into stadion

values(nr.nextval,10,'Maracana',70000);

insert into stadion

values(nr.nextval,11,'Arena de Sao Paulo',60000);

insert into stadion

values(nr.nextval,12,'Mineirao',55000);

insert into stadion

values(nr.nextval,13,'Estadio National de Brasilia',65000);

insert into stadion

values(nr.nextval,14,'Arena Castelaro',57000);

insert into stadion

values(nr.nextval,15,'Arena Fonte Navo',53000);

insert into stadion

values(nr.nextval,16,'Estadi Beira-Rio',42000);

insert into stadion

values(nr.nextval,17,'Arena Pernambuco',40000);

rollback;

--orase

insert into orase

values(10,'Rio de Janeiro',6780000,1200);

insert into orase

values (11,'Sao Paulo',12330000,1527);

insert into orase

values(12,'Belo Horizonte',2722000,330);

insert into orase

values(13,'Brasilia',2343000,5761);

insert into orase

values(14,'Fortaleza',2068000,318);

insert into orase

values(15,'Salvador',3920000,818);

insert into orase

values(16,'Porto Alegre',1500000,400);

insert into orase

values(17,'Recife',1200000,218);

--canale tv

insert into canale\_tv

values(1,'Digi Sport 1');

insert into canale\_tv

values(2,'Digi Sport 2');

insert into canale\_tv

values(3,'Digi Sport 3');

insert into canale\_tv

values(4,'Telekom Sport 1');

insert into canale\_tv

values(5,'Telekom Sport 2');

insert into canale\_tv

values(6,'Eurosport');

insert into canale\_tv

values(7,'Pro x');

--transmite AM RAMAS AIC

insert into transmite

values(1,1);

insert into transmite

values(1,4);

insert into transmite

values(1,8);

insert into transmite

values(1,10);

insert into transmite

values(1,12);

insert into transmite

values(1,13);

insert into transmite

values(1,15);

insert into transmite

values(2,2);

insert into transmite

values(2,5);

insert into transmite

values(2,6);

insert into transmite

values(2,9);

insert into transmite

values(2,14);

insert into transmite

values(3,3);

insert into transmite

values(3,7);

insert into transmite

values(3,11);

insert into transmite

values(4,1);

insert into transmite

values(4,4);

insert into transmite

values(4,8);

insert into transmite

values(4,10);

insert into transmite

values(4,12);

insert into transmite

values(4,13);

insert into transmite

values(4,15);

insert into transmite

values(5,2);

insert into transmite

values(5,5);

insert into transmite

values(5,6);

insert into transmite

values(5,9);

insert into transmite

values(5,14);

insert into transmite

values(6,3);

insert into transmite

values(6,7);

insert into transmite

values(6,11);

insert into transmite

values(7,13);

insert into transmite

values(8,13);

insert into transmite

values(7,14);

insert into transmite

values(8,14);

insert into transmite

values(7,15);

insert into transmite

values(8,15);

--antrenori

insert into antrenori

values(1,'Luiz Felipe','Scolari','BRA');

insert into antrenori

values(2,'Joachim','Low','GER');

insert into antrenori

values(3,'Vincente','del Bosque','SPA');

insert into antrenori

values(4,'Didider','Deschamps','FRA');

insert into antrenori

values(5,'Louis','Van Gaal','NED');

insert into antrenori

values(6,'Jorge','Sampaoli','CHI');

insert into antrenori

values(7,'Jose','Pekerman','COL');

insert into antrenori

values(8,'Jorge Luis','Pinto','CRC');

insert into antrenori

values(9,'Stephen','Kesi','NIG');

insert into antrenori

values(10,'Fernando','Santos','GRE');

insert into antrenori

values(11,'Oscar','Tabarez','URU');

insert into antrenori

values(12,'Ottmar','Hitzfeld','SUI');

insert into antrenori

values(13,'Jurgen','Kinsman','USA');

insert into antrenori

values(14,'Marc','Wilmots','BEL');

insert into antrenori

values(15,'Alejandro','Sabella','ARG');

insert into antrenori

values(16,'Miguel','Herrera','MEX');

--sponsorizata

insert into sponsorizeaza

values('ARG',200);

insert into sponsorizeaza

values('ARG',207);

insert into sponsorizeaza

values('ARG',204);

insert into sponsorizeaza

values('BRA',201);

insert into sponsorizeaza

values('BRA',202);

insert into sposnorizeaz

values('CHI',203);

insert into sponsorizeaza

values('NED',201);

insert into sponsorizeaza

values('NED',206);

insert into sponsorizeaza

values('FRA',201);

insert into sponsorizeaza

values('FRA',202);

insert into sponsorizeaza

values('FRA',208);

insert into sponsorizeaza

values('FRA',204);

insert into sponsorizeaza

values('SUI',201);

insert into sponsorizeaza

values('SUI',206);

insert into sponsorizeaza

values('CRC',203);

insert into sponsorizeaza

values('MEX',205);

insert into sponsorizeaza

values('GRE',203);

insert into sponsorizeaza

values('GER',200);

insert into sponsorizeaza

values('GER',206);

insert into sponsorizeaza

values('GER',209);

insert into sponsorizeaza

values('SPA',200);

insert into sponsorizeaza

values('SPA',207);

insert into sponsorizeaza

values('NIG',205);

insert into sponsorizeaza

values('ALG',205);

insert into sponsorizeaza

values('USA',200);

insert into sponsorizeaza

values('USA',203);

insert into sponsorizeaza

values('USA',209);

insert into sponsorizeaza

values('BEL',201);

insert into sponsorizeaza

values('BEL',204);

insert into sponsorizeaza

values('ENG',201);

insert into sponsorizeaza

values('ITA',200);

insert into sponsorizeaza

values('POR',201);

insert into sponsorizeaza

values('GHA',205);

insert into sponsorizeaza

values('CID',203);

insert into sponsorizeaza

values('RUS',203);

insert into sponsorizeaza

values('HON',205);

insert into sponsorizeaza

values('IRA',203);

insert into sponsorizeaza

values('URU',201);

insert into sponsorizeaza

values('JAP',200);

insert into sponsorizeaza

values('CRO',201);

insert into sponsorizeaza

values('CAM',200);

insert into sponsorizeaza

values('AUS',203);

create SEQUENCE nr

start with 1

increment by 1;

insert into meci

values(nr.nextval,'Optimi',Date'2014-7-8',8,'BRA','CHI',1,1,3,2);

insert into meci

values(nr.nextval,'Optimi',Date'2014-7-8',5,'COL','URU',2,0,null,null);

insert into meci

values(nr.nextval,'Optimi',Date'2014-7-9',7,'CRC','GRE',1,1,5,3);

insert into meci

values(nr.nextval,'Optimi',Date'2014-7-9',6,'GER','ALG',2,1,null,null);

insert into meci

values(nr.nextval,'Optimi',Date'2014-7-10',4,'FRA','NIG',2,0,null,null);

insert into meci

values(nr.nextval,'Optimi',Date'2014-7-10',3,'NED','MEX',2,1,null,null);

insert into meci

values(nr.nextval,'Optimi',Date'2014-7-11',2,'ARG','SUI',1,0,null,null);

insert into meci

values(nr.nextval,'Optimi',Date'2014-7-11',1,'BEL','USA',2,1,null,null);

insert into meci

values(nr.nextval,'Sfert',Date'2014-7-15',2,'FRA','GER',0,1,null,null);

insert into meci

values(nr.nextval,'Sfert',Date'2014-7-16',1,'BRA','COL',2,1,null,null);

insert into meci

values(nr.nextval,'Sfert',Date'2014-7-17',3,'ARG','BEL',1,0,NULL,NULL);

insert into meci

values(nr.nextval,'Sfert',Date'2014-7-18',4,'NED','CRC',0,0,4,3);

insert into meci

values(nr.nextval,'Semifinala',Date'2014-7-22',2,'NED','ARG',0,0,2,4);

insert into meci

values(nr.nextval,'Semifinala',Date'2014-7-23',3,'BRA','GER',1,7,NULL,NULL);

insert into meci

values(nr.nextval,'Finala',Date'2014-7-28',1,'GER','ARG',1,0,NULL,NULL);

rollback;

commit;

--bilete

insert into bilete

values(1000,'Standard\_Optimi');

insert into bilete

values(1001,'VIP\_Optimi');

insert into bilete

values(2000,'Standard\_Sferturi');

insert into bilete

values(2001,'VIP\_Sferturi');

insert into bilete

values(3000,'Standard\_Semi-finale');

insert into bilete

values(3001,'VIP\_Semi-finale');

insert into bilete

values(4000,'Standard-Finala');

insert into bilete

values(4001,'VIP-Finala');

--dispune

insert into dispune

values(1000,nr.nextval);

insert into dispune

values(1000,nr.nextval);

insert into dispune

values(1000,nr.nextval);

insert into dispune

values(1000,nr.nextval);

insert into dispune

values(1000,nr.nextval);

insert into dispune

values(1000,nr.nextval);

insert into dispune

values(1000,nr.nextval);

insert into dispune

values(1000,nr.nextval);

insert into dispune

values(2000,nr.nextval);

insert into dispune

values(2000,nr.nextval);

insert into dispune

values(2000,nr.nextval);

insert into dispune

values(2000,nr.nextval);

insert into dispune

values(3000,nr.nextval);

insert into dispune

values(3000,nr.nextval);

insert into dispune

values(4000,nr.nextval);

-------

drop sequence nr;

insert into dispune

values(1001,nr.nextval);

insert into dispune

values(1001,nr.nextval);

insert into dispune

values(1001,nr.nextval);

insert into dispune

values(1001,nr.nextval);

insert into dispune

values(1001,nr.nextval);

insert into dispune

values(1001,nr.nextval);

insert into dispune

values(1001,nr.nextval);

insert into dispune

values(1001,nr.nextval);

insert into dispune

values(2001,nr.nextval);

insert into dispune

values(2001,nr.nextval);

insert into dispune

values(2001,nr.nextval);

insert into dispune

values(2001,nr.nextval);

insert into dispune

values(3001,nr.nextval);

insert into dispune

values(3001,nr.nextval);

insert into dispune

values(4001,nr.nextval);

Cerinta 6

Pentru o etapa a meciurilor data selectati jucatorii din top 10 care joaca pe fiecare stadion

create or replace procedure p6(runda Meci.etapa%TYPE)

is

type t\_i is table of stadion%Rowtype index by pls\_integer;

stadioane t\_i;

type t is table of meci%rowtype;

meciuri t:=t();

type t\_i2 is table of jucatori%rowtype index by pls\_integer;

jucatori\_tab t\_i2;

numar number;

Begin

select count(\*)

into numar

from meci

where etapa=runda;

meciuri.extend(numar+1);

select\*

bulk collect into meciuri

from meci

where meci.etapa=runda;

select\*

bulk collect into stadioane

from stadion;

for i in meciuri.first..meciuri.last loop

for j in stadioane.first..stadioane.last loop

if (meciuri(i).id\_stadion=stadioane(j).id\_stadion)

then

DBMS\_OUTPUT.PUT\_LINE('In meciul '||i||' pe stadionul '||stadioane(j).stadion||'au jucat:');

select\*

bulk collect into jucatori\_tab

from jucatori

where id\_tara=meciuri(i).id\_echipa1 or id\_tara=meciuri(i).id\_echipa2;

if(jucatori\_tab.count!=0)

then

for z in jucatori\_tab.first..jucatori\_tab.last loop

DBMS\_OUTPUT.PUT\_LINE(jucatori\_tab(z).prenume||' '||jucatori\_tab(z).nume);

end loop;

else

dbms\_output.put\_line('niciunul');

end if;

end if;

end loop;

end loop;

End;

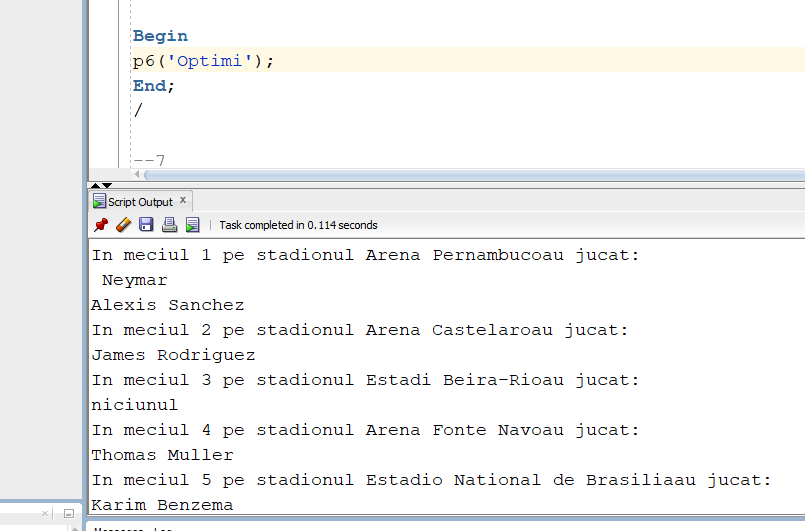
/

Begin

p6('Optimi');

End;

/



Cerinta 7

Afisati numarul de puncte al fiecarei echipe care a jucat in fazele eliminatorii impotriva unei echipe date ca parametru

create or replace procedure p7(echipa tari.id\_tara%TYPE)

is

cursor c (echipa tari.id\_tara%TYPE) is

select \*

from meci

where meci.id\_echipa1=echipa or echipa=meci.id\_echipa2;

cursor c1 (echipa Meci.id\_echipa1%TYPE) is

select numar\_puncte

from tari

where id\_tara=echipa;

cursor c2 (team Meci.id\_echipa2%TYPE) is

select nume\_tara

from tari

where tari.id\_tara=team;

puncte number;

team tari.id\_tara%TYPE;

nume tari.nume\_tara%TYPE;

Begin

for meci in c(echipa) LOOp

if meci.id\_echipa1=echipa

then

team:=meci.id\_echipa2;

else

team:=meci.id\_echipa1;

end if;

open c1(team);

fetch c1 into puncte;

open c2(team);

fetch c2 into nume;

DBMS\_OUTPUT.PUT\_LINE('Echipa '|| nume ||' a obtinut '|| puncte||' in faza grupelor');

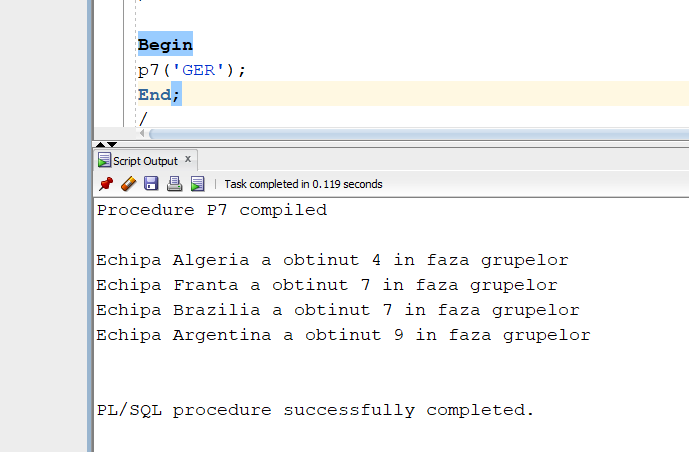
close c1;

close c2;

end loop;

End;

/



Cerinta 8

Afisati echipa cu cele mai multe puncte in faza grupelor care au un sponsor dat ca parametru si un numar minim de puncte acumulate.

create or replace function f8(sponsor sponsori.nume\_sponsor%TYPE,nr\_min number) return varchar2 is

tara\_max tari.nume\_tara%TYPE;

nr number;

max1 number:=nr\_min-1;

max2 number:=-1;

v number;

nume tari.nume\_tara%TYPE;

no\_country exception;

too\_many exception;

no\_data\_found exception;

Begin

select count(\*)

into v

from sponsori

where nume\_sponsor=sponsor;

if(v=0)

then

raise no\_data\_found;

end if;

for i in(select nume\_tara,numar\_puncte

from sponsorizeaza s1, sponsori s2, tari t

where sponsor=s2.nume\_sponsor and s2.id\_sponsor=s1.id\_sponsor and s1.id\_tara=t.id\_tara)

loop

nr:=i.numar\_puncte;

nume:=i.nume\_tara;

if(nr>max1)

then

max1:=nr;

tara\_max:=nume;

else

if(nr=max1)

then

max2:=nr;

end if;

end if;

end loop;

if(max1=nr\_min-1)

then

raise no\_country;

end if;

if(max1=max2)

then

raise too\_many;

end if;

return tara\_max;

Exception

when no\_country then

return('No such country!');

when too\_many then

return('More teams with same amount of points!');

when no\_data\_found then

return('No sponsor found');

End;

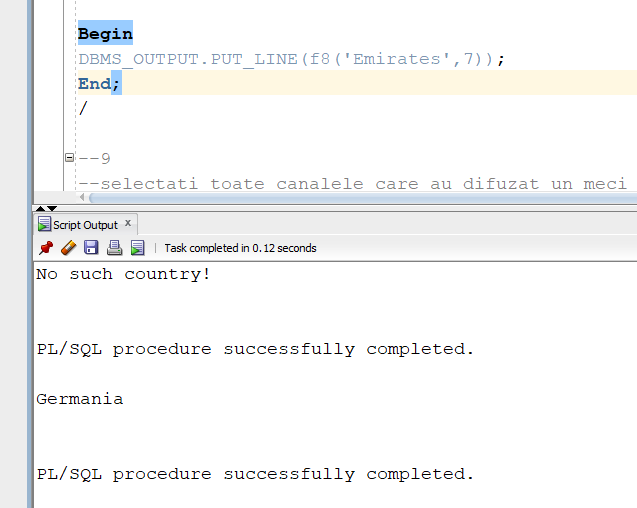
/

Begin

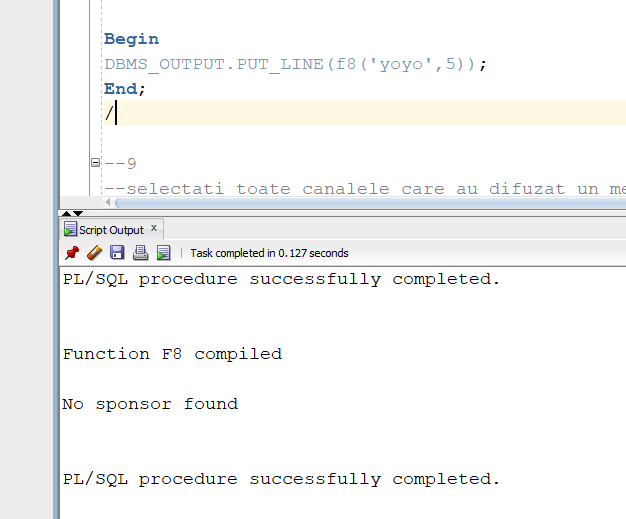
DBMS\_OUTPUT.PUT\_LINE(f8('Adidas',5));

End;

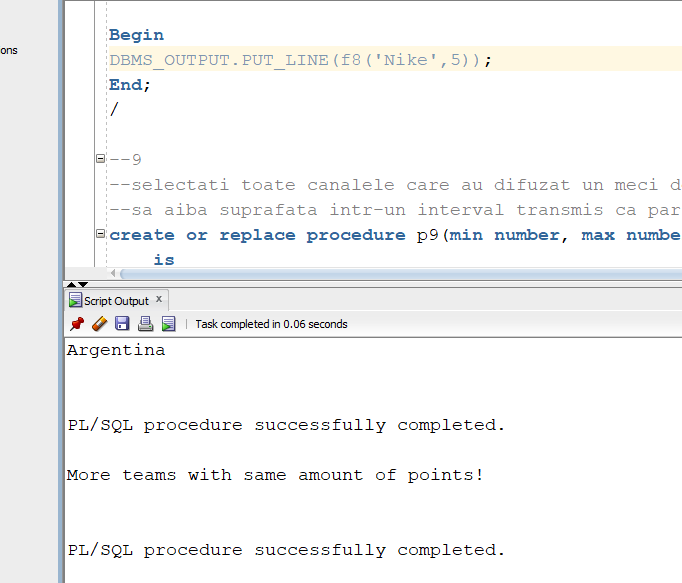
/



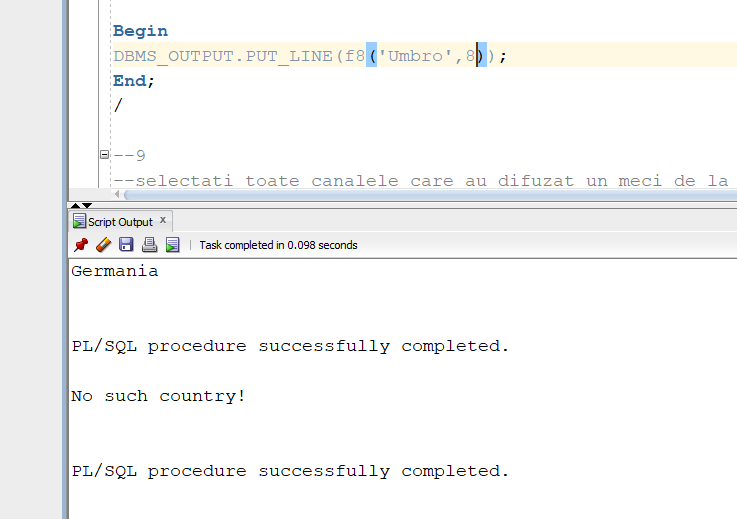
Exceptia no\_data\_found



Exception too\_many



Exception no\_country



Cerinta 9

Selectati toate canalele care au difuzat un meci de la sferturi in sus intr-un oras care sa aiba suprafata intr-un interval transmis ca parametru.

create or replace procedure p9(min number, max number)

is

nume\_oras orase.nume%TYPE;

Begin

select nume

into nume\_oras

from orase

where suprafata>=min and suprafata<=max;

DBMS\_OUTPUT.PUT\_LINE('Canale:');

for canal in(select nume\_canal

from canale\_tv c, transmite t,meci m,stadion s,orase o

where c.id\_canal=t.id\_canal and t.id\_meci=m.id\_meci and m.id\_meci>=9

and m.id\_stadion=s.id\_stadion and s.id\_oras=o.id\_oras and o.nume=nume\_oras)

loop

DBMS\_OUTPUT.PUT\_LINE(canal.nume\_canal);

end loop;

Exception

when No\_data\_found then

DBMS\_OUTPUT.PUT\_LINE('Nu exista oras cu suprafata ceruta');

when too\_many\_rows then

DBMS\_OUTPUT.PUT\_LINE('Prea multe orase cu suprafata ceruta');

End;

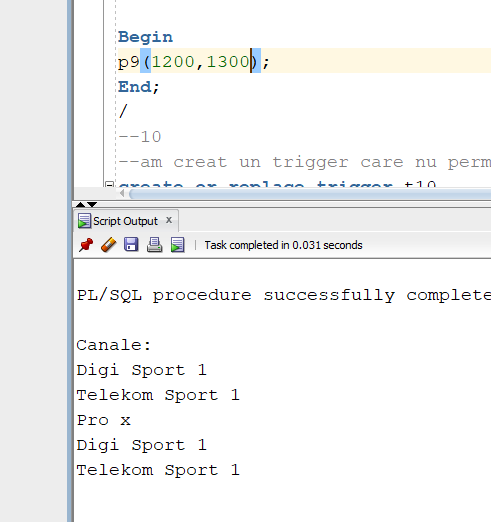
/

Begin

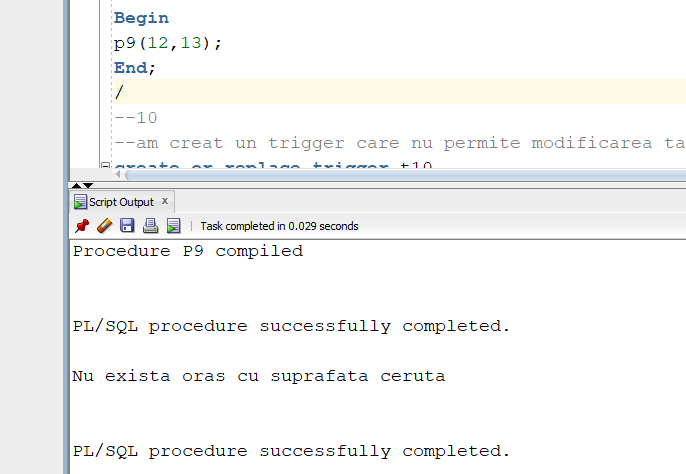
p9(1200,1300);

End;

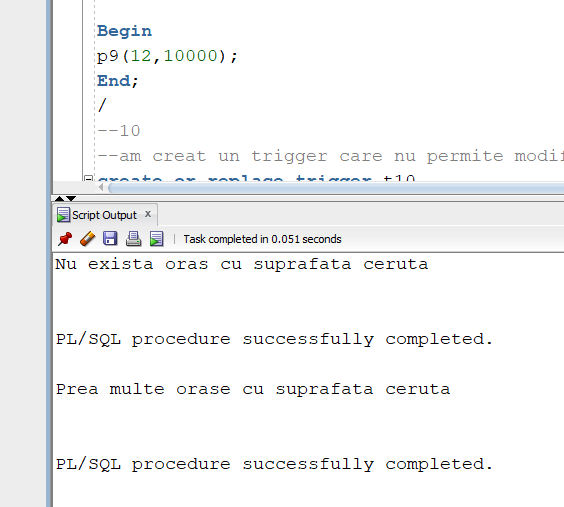
/



Exception no\_data\_found



Exception too\_many\_rows



Cerinta 10

Am creat un trigger care nu permite modificarea tabele meci in afara campionatului mondial

create or replace trigger t10

before insert or update or delete on meci

Begin

if(sysdate>'01/AUG/2014' or sysdate<'01/JUL/2014')

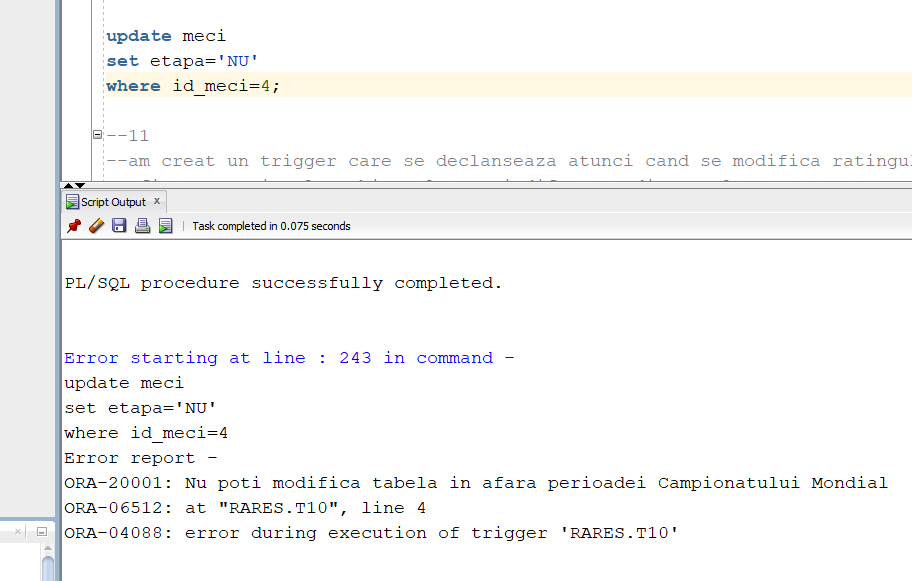
then

raise\_application\_error(-20001,'Nu poti modifica tabela in afara perioadei Campionatului Mondial');

end if;

End;

/



Cerinta 11

Am creat un trigger care se declanseaza atunci cand se modifica ratingul unui jucator si afiseaza ratingul vechi, cel nou si diferenta dintre ele

create or replace trigger t11

before insert or update or delete on jucatori

for each row

Declare

dif number;

begin

dif:=:New.rating-:Old.rating;

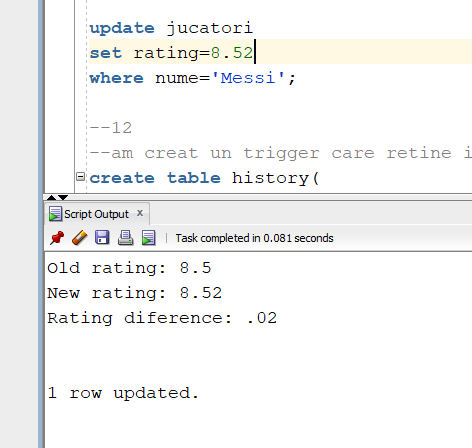
DBMS\_OUTPUT.PUT\_LINE('Old rating: '||:old.rating);

DBMS\_OUTPUT.PUT\_LINE('New rating: '||:new.rating);

DBMS\_OUTPUT.PUT\_LINE('Rating diference: '||dif);

End;

/



Cerinta 12

Am creat un trigger care retine in tabela history istoricul evenimentelor

create table history(

object\_type varchar(20),

object\_name varchar2(30),

event varchar2(30)

);

create or replace trigger t12

before create or alter or drop on schema

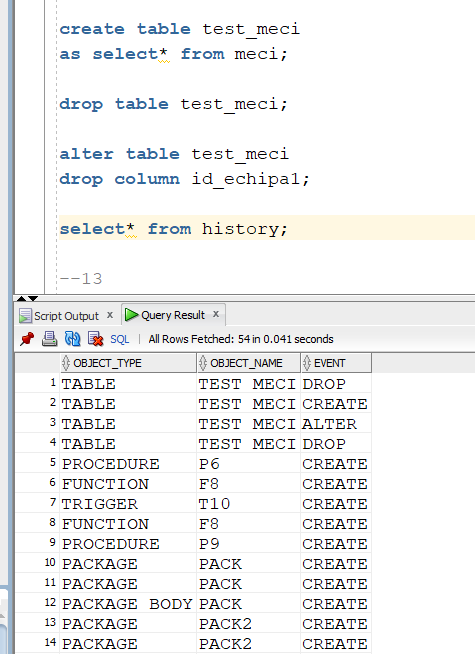
Begin

INSERT into history

values(SYS.dictionary\_obj\_type,SYS.dictionary\_obj\_name,SYS.sysevent);

End;

/



Cerinta 13

create or replace package pack

is

procedure p6(runda Meci.etapa%TYPE);

procedure p7(echipa tari.id\_tara%TYPE);

function f8(sponsor sponsori.nume\_sponsor%TYPE,nr\_min number) return varchar2;

procedure p9(min number, max number);

End pack;

/

create or replace package body pack

is

procedure p6(runda Meci.etapa%TYPE)

is

type t\_i is table of stadion%Rowtype index by pls\_integer;

stadioane t\_i;

type t is table of meci%rowtype;

meciuri t:=t();

type t\_i2 is table of jucatori%rowtype index by pls\_integer;

jucatori\_tab t\_i2;

numar number;

Begin

select count(\*)

into numar

from meci

where etapa=runda;

meciuri.extend(numar+1);

select\*

bulk collect into meciuri

from meci

where meci.etapa=runda;

select\*

bulk collect into stadioane

from stadion;

for i in meciuri.first..meciuri.last loop

for j in stadioane.first..stadioane.last loop

if (meciuri(i).id\_stadion=stadioane(j).id\_stadion)

then

DBMS\_OUTPUT.PUT\_LINE('In meciul '||i||' pe stadionul '||stadioane(j).stadion||'au jucat:');

select\*

bulk collect into jucatori\_tab

from jucatori

where id\_tara=meciuri(i).id\_echipa1 or id\_tara=meciuri(i).id\_echipa2;

if(jucatori\_tab.count!=0)

then

for z in jucatori\_tab.first..jucatori\_tab.last loop

DBMS\_OUTPUT.PUT\_LINE(jucatori\_tab(z).prenume||' '||jucatori\_tab(z).nume);

end loop;

else

dbms\_output.put\_line('niciunul');

end if;

end if;

end loop;

end loop;

End p6;

--7

--Afisati numarul de puncte al fiecarei echipe care a jucat in fazele eliminatorii impotriva unei echipe date ca parametru

procedure p7(echipa tari.id\_tara%TYPE)

is

cursor c (echipa tari.id\_tara%TYPE) is

select \*

from meci

where meci.id\_echipa1=echipa or echipa=meci.id\_echipa2;

cursor c1 (echipa Meci.id\_echipa1%TYPE) is

select numar\_puncte

from tari

where id\_tara=echipa;

cursor c2 (team Meci.id\_echipa2%TYPE) is

select nume\_tara

from tari

where tari.id\_tara=team;

puncte number;

team tari.id\_tara%TYPE;

nume tari.nume\_tara%TYPE;

Begin

for meci in c(echipa) LOOp

if meci.id\_echipa1=echipa

then

team:=meci.id\_echipa2;

else

team:=meci.id\_echipa1;

end if;

open c1(team);

fetch c1 into puncte;

open c2(team);

fetch c2 into nume;

DBMS\_OUTPUT.PUT\_LINE('Echipa '|| nume ||' a obtinut '|| puncte||' in faza grupelor');

close c1;

close c2;

end loop;

End p7;

--8

--Afisati echipa cu cele mai multe puncte in faza grupelor care au un sponsor dat ca parametru si un numar minim de puncte acumulate

function f8(sponsor sponsori.nume\_sponsor%TYPE,nr\_min number) return varchar2 is

tara\_max tari.nume\_tara%TYPE;

nr number;

max1 number:=nr\_min-1;

max2 number:=-1;

v number;

nume tari.nume\_tara%TYPE;

no\_country exception;

too\_many exception;

no\_data\_found exception;

Begin

select count(\*)

into v

from sponsori

where nume\_sponsor=sponsor;

if(v=0)

then

raise no\_data\_found;

end if;

for i in(select nume\_tara,numar\_puncte

from sponsorizeaza s1, sponsori s2, tari t

where sponsor=s2.nume\_sponsor and s2.id\_sponsor=s1.id\_sponsor and s1.id\_tara=t.id\_tara)

loop

nr:=i.numar\_puncte;

nume:=i.nume\_tara;

if(nr>max1)

then

max1:=nr;

tara\_max:=nume;

else

if(nr=max1)

then

max2:=nr;

end if;

end if;

end loop;

if(max1=nr\_min-1)

then

raise no\_country;

end if;

if(max1=max2)

then

raise too\_many;

end if;

return tara\_max;

Exception

when no\_country then

return('No such country!');

when too\_many then

return('More teams with same amount of points!');

when no\_data\_found then

return('No sponsor found');

End f8;

--9

--selectati toate canalele care au difuzat un meci de la sferturi in sus intr-un oras care

--sa aiba suprafata intr-un interval transmis ca parametru

procedure p9(min number, max number)

is

nume\_oras orase.nume%TYPE;

Begin

select nume

into nume\_oras

from orase

where suprafata>=min and suprafata<=max;

DBMS\_OUTPUT.PUT\_LINE('Canale:');

for canal in(select nume\_canal

from canale\_tv c, transmite t,meci m,stadion s,orase o

where c.id\_canal=t.id\_canal and t.id\_meci=m.id\_meci and m.id\_meci>=9

and m.id\_stadion=s.id\_stadion and s.id\_oras=o.id\_oras and o.nume=nume\_oras)

loop

DBMS\_OUTPUT.PUT\_LINE(canal.nume\_canal);

end loop;

Exception

when No\_data\_found then

DBMS\_OUTPUT.PUT\_LINE('Nu exista oras cu suprafata ceruta');

when too\_many\_rows then

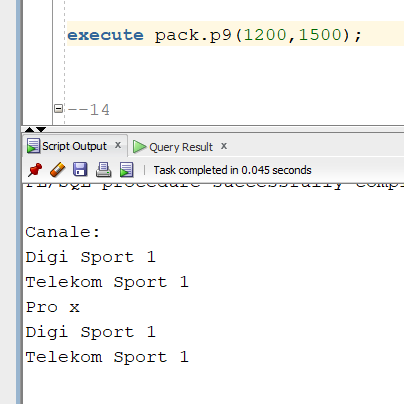
DBMS\_OUTPUT.PUT\_LINE('Prea multe orase cu suprafata ceruta');

End p9;

End pack;

/

execute pack.p9(1200,1500);



Cerinta 14

Pentru o tara introdusa de la tastatura sa se afiseze numarul de puncte, numarul de victorii din fazele eliminatorii, cati jucatori are in top 10

create or replace package pack2

is

type t\_i is table of tari%Rowtype index by pls\_integer;

tari\_table t\_i;

type t is table of meci%rowtype;

meciuri t:=t();

function get\_points(id\_tara tari.id\_tara%TYPE) return number;

function get\_id(nume tari.nume\_tara%TYPE) return tari.id\_tara%TYPE;

procedure top\_10(id\_tara tari.id\_tara%TYPE);

procedure victorii(id\_tara tari.id\_tara%TYPE);

procedure get\_details(nume tari.nume\_tara%TYPE);

END pack2;

/

Create or replace package body pack2

is

function get\_id(nume tari.nume\_tara%TYPE) return tari.id\_tara%TYPE

is

id1 tari.id\_tara%TYPE;

begin

select id\_tara

into id1

from tari

where nume\_tara=nume;

return id1;

Exception

when no\_data\_found then

DBMS\_OUTPUT.PUT\_LINE('Nu exista tara');

return -1;

end get\_id;

function get\_points(id\_tara tari.id\_tara%TYPE) return number

is

begin

select\*

bulk collect into tari\_table

from tari;

for i in tari\_table.first..tari\_table.last loop

if tari\_table(i).id\_tara=id\_tara

then

return tari\_table(i).numar\_puncte;

end if;

end loop;

end get\_points;

procedure top\_10(id\_tara tari.id\_tara%TYPE)

is

nr number:=0;

begin

for j in(select\* from jucatori) loop

for i in tari\_table.first..tari\_table.last loop

if(j.id\_tara=tari\_table(i).id\_tara and id\_tara=tari\_table(i).id\_tara)

then

nr:=nr+1;

end if;

end loop;

end loop;

if(nr=0)

then

DBMS\_OUTPUT.PUT\_LINE('Nu are jucatori in top 10');

else

DBMS\_OUTPUT.PUT\_LINE('Are '||nr||' jucatori in top 10');

end if;

end top\_10;

procedure victorii(id\_tara tari.id\_tara%TYPE)

is

nr number:=0;

victorii number:=0;

begin

select count(\*)

into nr

from meci;

meciuri.extend(nr+1);

select\*

bulk collect into meciuri

from meci;

for i in meciuri.first..meciuri.last loop

if(meciuri(i).id\_echipa1=id\_tara)

then

if(meciuri(i).goluri1>meciuri(i).goluri2)

then

victorii:=victorii+1;

else

if(meciuri(i).goluri1=meciuri(i).goluri2)

then

if(meciuri(i).penaltiuri1>meciuri(i).penaltiuri2)

then

victorii:=victorii+1;

end if;

end if;

end if;

end if;

if(meciuri(i).id\_echipa2=id\_tara)

then

if(meciuri(i).goluri1<meciuri(i).goluri2)

then

victorii:=victorii+1;

else

if(meciuri(i).goluri1=meciuri(i).goluri2)

then

if(meciuri(i).penaltiuri1<meciuri(i).penaltiuri2)

then

victorii:=victorii+1;

end if;

end if;

end if;

end if;

end loop;

DBMS\_OUTPUT.PUT\_LINE('Are '||victorii||' victorii');

end victorii;

procedure get\_details(nume tari.nume\_tara%TYPE)

is

puncte number;

id\_tara tari.id\_tara%TYPE;

begin

id\_tara:=get\_id(nume);

puncte:=get\_points(id\_tara);

DBMS\_OUTPUT.PUT\_LINE(nume||' a obtinut '||puncte||' puncte si:');

victorii(id\_tara);

top\_10(id\_tara);

end get\_details;

end pack2;

/

execute pack2.get\_details('Germania');

