

Proiect SGBD

Gestiunea Competitiilor de Sarituri cu Schiurile editia 2022-2023

Gheorghe Cosmina

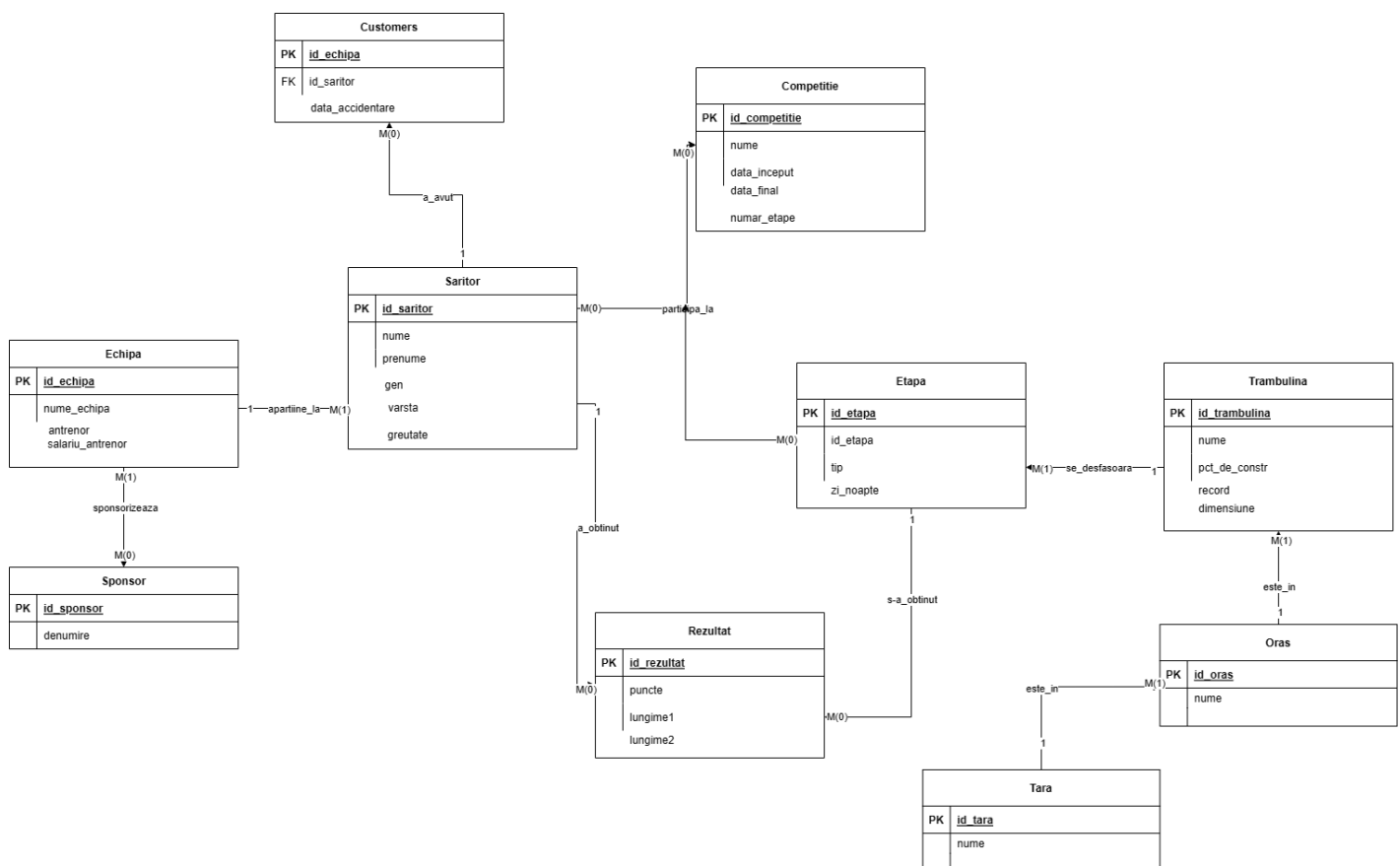
Gr 231

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

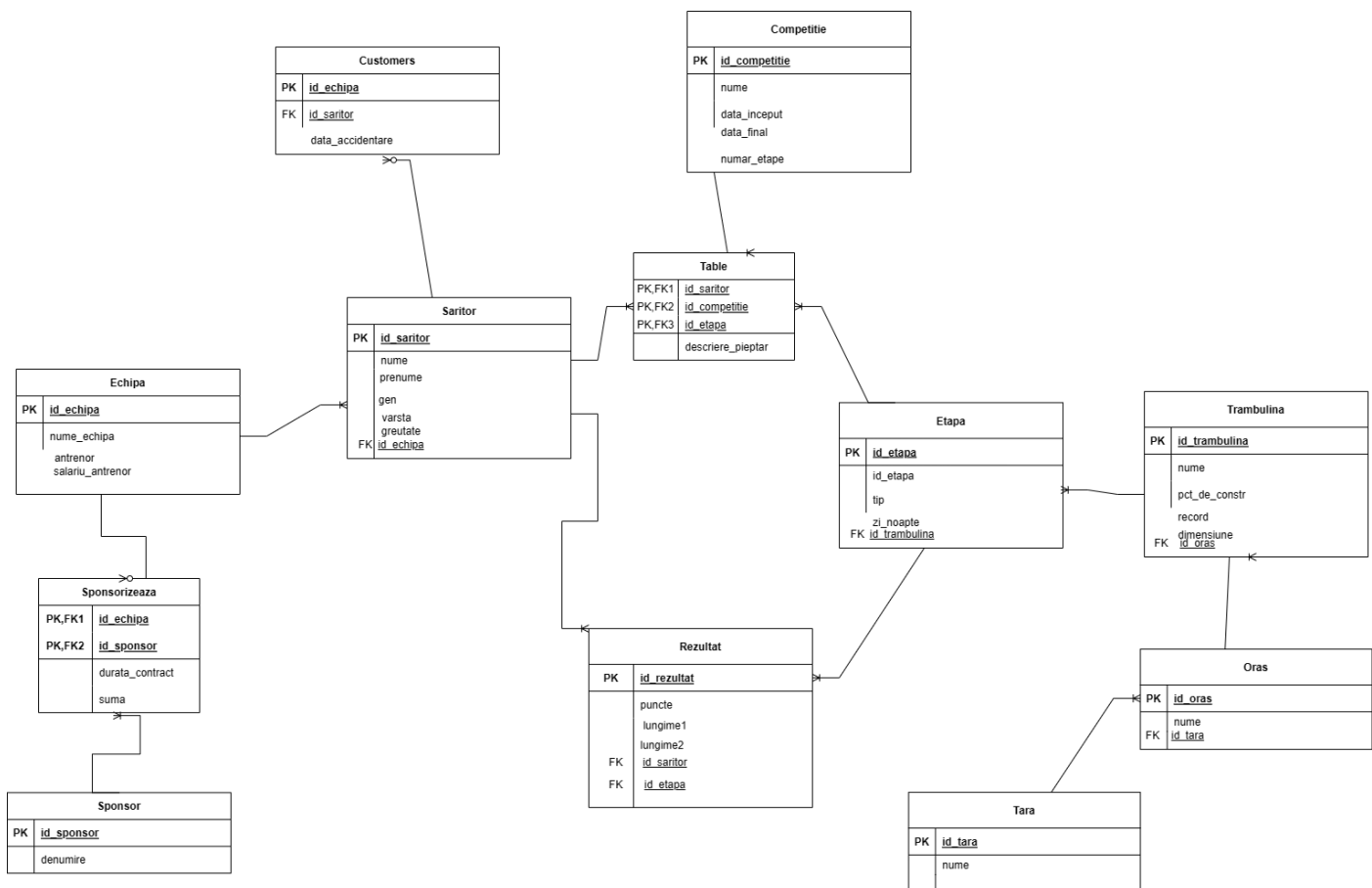
În această bază de date reținem informații despre ediția de sărituri cu schiurile din 2022-2023. Voi avea detalii despre săritori, accidentări, echipe, sponsori, etape, competiții, rezultate, dar și despre trambulinele, orașele și țările în care se desfășoară etapele. Un săritor poate aparține de o singură echipă, poate participa la mai multe etape și la mai multe competiții, o etapă putând face parte din mai multe competiții.

Spre exemplu, un săritor poate alege să participe la o etapă din cadrul competiției mondiale dar să nu se înscrie și la turneul RawAir deși etapa respectivă face parte atât din mondial cât și din RawAir.

2. Realizați diagrama entitate-relație (ERD).



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



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

--creare

```

create table ECHIPACGH(
id_echipa varchar(3) not null primary key,
nume_echipa varchar(25) not null,
antrenor varchar(50) not null,
salariu_antrenor numeric(4,0) check(salariu_antrenor>2000)
);
commit;
  
```

```

create table SARITORCGH
(
id_saritor number(5,0) not null primary key,
nume varchar(30) not null,
prenume varchar(30) not null,
gen varchar(1),
varsta number(2,0) not null,
greutate number(3,1) not null,
id_echipa varchar(3) not null,
foreign key (id_echipa) references ECHIPACGH(id_echipa) on delete cascade
  
```

);

```
create table ACCIDENTARECGH(  
id_accident number(3,0) not null primary key,  
id_saritor number(5,0) not null,  
data_accidentare DATE,  
foreign key (id_saritor) references SARITORCGH(id_saritor) on delete cascade  
);
```

```
create table SPONSORCGH(  
id_sponsor number(5,0) not null primary key,  
denumire varchar(40) not null  
);
```

```
create table SPONSORIZEAZACGH(  
id echipa varchar(5) not null,  
id_sponsor number(5,0) not null,  
durata_contract number(2,0) not null,  
suma number(4,0) not null,  
constraint pk_sponsorizeaza primary key(id_sponsor,id_echipa)  
);
```

```
create table COMPETITIECGH(  
id_competitie number(7,0) not null primary key,  
nume varchar(45) not null,  
data_inceput DATE,  
data_final DATE,  
numar_etape number(2,0) not null  
);
```

```
create table TARACGH(  
id_tara number(2,0) not null primary key,  
nume varchar(15) not null  
);
```

```
create table ORASCGH(  
id_oras number(3,0) not null primary key,  
id_tara number(2,0) not null,  
nume varchar(15) not null,  
foreign key (id_tara) references TARACGH(id_tara) on delete cascade  
);
```

```
create table TRAMBULINACGH(  
id_trambulina number(3,0) not null primary key,  
id_oras number(3,0) not null,  
nume varchar(40) not null,  
pct_de_constr number(3,0) not null,  
record number(4,1),  
dimensiune number(3,0) not null,  
foreign key (id_oras) references ORASCGH(id_oras) on delete cascade  
);
```

```
create table ETAPACGH (
```

```

id_trambulina number(3,0) not null,
id_etapa number(3,0) not null primary key,
data_etapa DATE,
tip varchar(9) not null,
zi_noapte varchar(7) not null,
foreign key (id_trambulina) references TRAMBULINACGH(id_trambulina) on delete cascade
);
commit;

```

```

create table PARTICIPA_LACGH (
id_etapa number(3,0) not null,
id_competitie number(7,0) not null,
id_saritor number(5,0) not null,
descriere_pieptar varchar(50),
constraint pk_participa_la primary key(id_competitie,id_saritor,id_etapa)
);

```

```

create table REZULTATCGH(
id_saritor number(5,0) not null,
id_etapa number(3,0) not null,
id_rezultat number(5,0) not null,
puncte number(5,2)not null,
lungime1 number(5,1),
lungime2 number(5,1),
constraint pk_rezultat primary key(id_etapa,id_rezultat,id_saritor)
);

```

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

```

--inserare
insert into ECHIPACGH values
('NOV','Norvegia','Roar Ljokelsoy',2300);
insert into ECHIPACGH values
('SLO','Slovenia','Abi Lanon',4300);
insert into ECHIPACGH values
('GER','Germania','Pius Pashke',2900);
insert into ECHIPACGH values
('AUT','Austria','Sven Ruperdadt',4600);
insert into ECHIPACGH values
('ROU','Romania','Dumitru Petre',2400);

insert into SARITORCGH values
('401','Kraft','Stefan','m',29,56,'AUT');
insert into SARITORCGH values
('402','Horl','Jan','m',23,67,'AUT');
insert into SARITORCGH values
('403','Huber','Daniel','m',25,64,'AUT');
insert into SARITORCGH values
('404','Morgenstern','Thomas','m',40,66,'AUT');
insert into SARITORCGH values

```

('100','Eisenbichler','Markus','m',29,56,'GER');
insert into SARITORCGH values
('101','Geiger','Karl','m',23,67,'GER');
insert into SARITORCGH values
('102','Wellinger','Andreas','m',25,64,'GER');
insert into SARITORCGH values
('301','Kos','Lovro','m',23,67,'SLO');
insert into SARITORCGH values
('302','Prevc','Peter','m',25,64,'SLO');
insert into SARITORCGH values
('201','Tande','Daniel-Andre','m',23,67,'NOV');

insert into SPONSORCGH values
(123,'milka');
insert into SPONSORCGH values
(789,'ck');
insert into SPONSORCGH values
(456,'4F');
insert into SPONSORCGH values
(135,'adidas');
insert into SPONSORCGH values
(246,'nike');

insert into SPONSORIZEAZACGH values
('SLO',135,24,5700);
insert into SPONSORIZEAZACGH values
('ROU',789,6,3300);
insert into SPONSORIZEAZACGH values
('AUT',135,12,5700);
insert into SPONSORIZEAZACGH values
('NOV',135,18,4700);
insert into SPONSORIZEAZACGH values
('ROU',135,9,4700);
insert into SPONSORIZEAZACGH values
('NOV',456,6,3000);
insert into SPONSORIZEAZACGH values
('NOV',246,12,3400);
insert into SPONSORIZEAZACGH values
('GER',246,12,4400);
insert into SPONSORIZEAZACGH values
('GER',135,6,2700);
insert into SPONSORIZEAZACGH values
('GER',123,12,4600);

insert into COMPETITIECGH values
(56789, 'Zbor cu schiurile','02-DEC-2022','05-DEC-2023',2);
insert into COMPETITIECGH values
(34567, 'RAW AIR','09-MAR-2023','18-MAR-2023',5);
insert into COMPETITIECGH values
(45678, 'Planica 3','21-FEB-2023','05-MAR-2023',3);
insert into COMPETITIECGH values
(12345, 'Mondial','21-NOV-2022','18-MAR-2023',12);
insert into COMPETITIECGH values

(23456, 'Extra', '05-DEC-23', '05-DEC-23', 1);

insert into TARACGH values

(4, 'Slovenia');

insert into TARACGH values

(1, 'Polonia');

insert into TARACGH values

(3, 'Germania');

insert into TARACGH values

(2, 'Austria');

insert into TARACGH values

(5, 'Norvegia');

insert into ORASCGH values

(11, 1, 'Zakopane');

insert into ORASCGH values

(31, 3, 'Oberstdorf');

insert into ORASCGH values

(32, 3, 'Klingenthal');

insert into ORASCGH values

(21, 2, 'Innsbruck');

insert into ORASCGH values

(51, 5, 'Vikersund');

insert into ORASCGH values

(41, 4, 'Planica');

insert into TRAMBULINACGH values

(211, 21, 'Berg Isel', 120, 134.5, 130);

insert into TRAMBULINACGH values

(212, 21, 'Paul-Ausserleitner-Schanze', 125, 145, 140);

insert into TRAMBULINACGH values

(311, 31, 'Heini-Klopfer-Skilflugschanze', 185, 225.5, 213);

insert into TRAMBULINACGH values

(321, 32, 'Olympiaschanze', 125, 141, 140);

insert into TRAMBULINACGH values

(411, 41, 'Letalnica', 185, 239, 215);

insert into TRAMBULINACGH values

(511, 51, 'Vikersundbakken', 200, 254, 240);

commit;

insert into ETAPACGH values

(212, 3, '2-DEC-2022', 'zbor', 'noapte');

insert into ETAPACGH values

(211, 4, '24-DEC-2022', 'sarit', 'zi');

insert into ETAPACGH values

(311, 8, '05-DEC-2022', 'sarit', 'zi');

insert into ETAPACGH values

(321, 12, '04-JAN-2023', 'sarit', 'zi');

insert into ETAPACGH values

```
(411,14,'27-FEB-2023','sarit','noapte');
insert into ETAPACGH values
(212,11,'23-FEB-2023','zbor','noapte');
insert into ETAPACGH values
(411,15,'05-MAR-2023','sarit','zi');
insert into ETAPACGH values
(311,16,'09-MAR-2023','sarit','zi');
insert into ETAPACGH values
(311,17,'10-MAR-2023','sarit','zi');
insert into ETAPACGH values
(511,19,'16-MAR-2023','sarit','noapte');
insert into ETAPACGH values
(511,20,'17-MAR-2023','sarit','noapte');
insert into ETAPACGH values
(511,21,'18-MAR-2023','sarit','zi');

commit;
```

```
insert into PARTICIPA_LACGH values
(3,12345,301,'galben');
insert into PARTICIPA_LACGH values
(3,12345,401,'galben');
insert into PARTICIPA_LACGH values
(3,12345,402,'galben');
insert into PARTICIPA_LACGH values
(12,12345,102,'mov');
insert into PARTICIPA_LACGH values
(12,12345,201,'mov');
insert into PARTICIPA_LACGH values
(12,12345,301,'mov');
insert into PARTICIPA_LACGH values
(12,12345,302,'mov');
insert into PARTICIPA_LACGH values
(12,12345,401,'mov');
insert into PARTICIPA_LACGH values
(12,12345,402,'mov');
insert into PARTICIPA_LACGH values
(12,12345,403,'mov');
insert into PARTICIPA_LACGH values
(12,12345,404,'mov');
insert into PARTICIPA_LACGH values
(12,12345,100,'mov');
insert into PARTICIPA_LACGH values
(3,56789,100,'verde');
insert into PARTICIPA_LACGH values
(3,56789,101,'verde');
insert into PARTICIPA_LACGH values
(3,56789,102,'verde');
insert into PARTICIPA_LACGH values
(3,56789,201,'verde');
insert into PARTICIPA_LACGH values
(3,56789,301,'verde');
```


insert into PARTICIPA_LACGH values
(3,56789,302,'verde');
insert into PARTICIPA_LACGH values
(3,56789,401,'verde');
insert into PARTICIPA_LACGH values
(3,56789,402,'verde');
insert into PARTICIPA_LACGH values
(3,56789,403,'verde');
insert into PARTICIPA_LACGH values
(3,56789,404,'verde');
insert into PARTICIPA_LACGH values
(3,56789,501,'verde');
insert into PARTICIPA_LACGH values
(8,56789,100,'alb');
insert into PARTICIPA_LACGH values
(8,56789,101,'alb');
insert into PARTICIPA_LACGH values
(8,56789,102,'alb');
insert into PARTICIPA_LACGH values
(8,56789,201,'alb');
insert into PARTICIPA_LACGH values
(8,56789,301,'alb');
insert into PARTICIPA_LACGH values
(8,56789,302,'alb');
insert into PARTICIPA_LACGH values
(8,56789,401,'alb');
insert into PARTICIPA_LACGH values
(8,56789,402,'alb');
insert into PARTICIPA_LACGH values
(8,56789,403,'alb');
insert into PARTICIPA_LACGH values
(8,56789,404,'alb');
insert into PARTICIPA_LACGH values
(8,56789,501,'alb');

insert into PARTICIPA_LACGH values
(17,34567,301,'portocaliu');
insert into PARTICIPA_LACGH values
(17,34567,401,'portocaliu');
insert into PARTICIPA_LACGH values
(17,34567,402,'portocaliu');
insert into PARTICIPA_LACGH values
(19,34567,102,'roz');
insert into PARTICIPA_LACGH values
(19,34567,201,'roz');
insert into PARTICIPA_LACGH values
(19,34567,301,'roz');
insert into PARTICIPA_LACGH values
(19,34567,302,'roz');
insert into PARTICIPA_LACGH values
(19,34567,401,'roz');
insert into PARTICIPA_LACGH values
(19,34567,402,'roz');

insert into PARTICIPA_LACGH values
(19,34567,403,'roz');
insert into PARTICIPA_LACGH values
(19,34567,404,'roz');
insert into PARTICIPA_LACGH values
(19,34567,100,'roz');
insert into PARTICIPA_LACGH values
(11,45678,100,'maro');
insert into PARTICIPA_LACGH values
(11,45678,101,'maro');
insert into PARTICIPA_LACGH values
(11,45678,102,'maro');
insert into PARTICIPA_LACGH values
(11,45678,201,'maro');
insert into PARTICIPA_LACGH values
(11,45678,301,'maro');
insert into PARTICIPA_LACGH values
(11,45678,302,'maro');
insert into PARTICIPA_LACGH values
(11,45678,401,'maro');
insert into PARTICIPA_LACGH values
(11,45678,402,'maro');
insert into PARTICIPA_LACGH values
(11,45678,403,'maro');
insert into PARTICIPA_LACGH values
(11,45678,404,'maro');
insert into PARTICIPA_LACGH values
(11,45678,501,'maro');

insert into PARTICIPA_LACGH values
(3,12345,100,'verde');
insert into PARTICIPA_LACGH values
(3,12345,101,'verde');
insert into PARTICIPA_LACGH values
(3,12345,102,'verde');
insert into PARTICIPA_LACGH values
(3,12345,201,'verde');
insert into PARTICIPA_LACGH values
(3,12345,302,'verde');
insert into PARTICIPA_LACGH values
(3,12345,403,'verde');
insert into PARTICIPA_LACGH values
(3,12345,404,'verde');
insert into PARTICIPA_LACGH values
(3,12345,501,'verde');
insert into PARTICIPA_LACGH values
(8,12345,100,'alb');
insert into PARTICIPA_LACGH values
(8,12345,101,'alb');
insert into PARTICIPA_LACGH values
(8,12345,102,'alb');
insert into PARTICIPA_LACGH values
(8,12345,201,'alb');

insert into PARTICIPA_LACGH values
(8,12345,301,'alb');
insert into PARTICIPA_LACGH values
(8,12345,302,'alb');
insert into PARTICIPA_LACGH values
(8,12345,401,'alb');
insert into PARTICIPA_LACGH values
(8,12345,402,'alb');
insert into PARTICIPA_LACGH values
(8,12345,403,'alb');
insert into PARTICIPA_LACGH values
(8,12345,404,'alb');
insert into PARTICIPA_LACGH values
(8,12345,501,'alb');

insert into PARTICIPA_LACGH values
(17,12345,301,'portocaliu');
insert into PARTICIPA_LACGH values
(17,12345,401,'portocaliu');
insert into PARTICIPA_LACGH values
(17,12345,402,'portocaliu');
insert into PARTICIPA_LACGH values
(19,12345,102,'roz');
insert into PARTICIPA_LACGH values
(19,12345,201,'roz');
insert into PARTICIPA_LACGH values
(19,12345,301,'roz');
insert into PARTICIPA_LACGH values
(19,12345,302,'roz');
insert into PARTICIPA_LACGH values
(19,12345,401,'roz');
insert into PARTICIPA_LACGH values
(19,12345,402,'roz');
insert into PARTICIPA_LACGH values
(19,12345,403,'roz');
insert into PARTICIPA_LACGH values
(19,12345,404,'roz');
insert into PARTICIPA_LACGH values
(19,12345,100,'roz');
insert into PARTICIPA_LACGH values
(11,12345,100,'maro');
insert into PARTICIPA_LACGH values
(11,12345,101,'maro');
insert into PARTICIPA_LACGH values
(11,12345,102,'maro');
insert into PARTICIPA_LACGH values
(11,12345,201,'maro');
insert into PARTICIPA_LACGH values
(11,12345,301,'maro');
insert into PARTICIPA_LACGH values
(11,12345,302,'maro');
insert into PARTICIPA_LACGH values
(11,12345,401,'maro');

```
insert into PARTICIPA_LACGH values  
(11,12345,402,'maro');  
insert into PARTICIPA_LACGH values  
(11,12345,403,'maro');  
insert into PARTICIPA_LACGH values  
(11,12345,404,'maro');  
insert into PARTICIPA_LACGH values  
(11,12345,501,'maro');
```

```
insert into PARTICIPA_LACGH values  
(8,23456,100,'alb');  
insert into PARTICIPA_LACGH values  
(8,23456,101,'alb');  
insert into PARTICIPA_LACGH values  
(8,23456,102,'alb');  
insert into PARTICIPA_LACGH values  
(8,23456,201,'alb');  
insert into PARTICIPA_LACGH values  
(8,23456,301,'alb');  
insert into PARTICIPA_LACGH values  
(8,23456,302,'alb');  
insert into PARTICIPA_LACGH values  
(8,23456,401,'alb');  
insert into PARTICIPA_LACGH values  
(8,23456,402,'alb');  
insert into PARTICIPA_LACGH values  
(8,23456,403,'alb');  
insert into PARTICIPA_LACGH values  
(8,23456,404,'alb');
```

```
commit;
```

```
insert into ACCIDENTARECGH values  
(777,302,'05-DEC-2022');  
insert into ACCIDENTARECGH values  
(111,100,'05-DEC-2022');  
insert into ACCIDENTARECGH values  
(222,302,'23-FEB-2023');  
insert into ACCIDENTARECGH values  
(333,301,'23-FEB-2023');  
insert into ACCIDENTARECGH values  
(444,402,'10-MAR-2023');
```

```
insert into RESULTATCGH values  
(301,3,31,220,150,150);  
insert into RESULTATCGH values  
(401,3,32,210,143,147);  
insert into RESULTATCGH values  
(402,3,33,212,142,150);
```

```
insert into RESULTATCGH values
```

(201,12,121,200,137,140);
insert into RESULTATCGH values
(102,12,122,210,143,145);
insert into RESULTATCGH values
(302,12,123,212,132,110);
insert into RESULTATCGH values
(301,12,124,200,117,130);
insert into RESULTATCGH values
(401,12,125,210,123,115);
insert into RESULTATCGH values
(100,12,126,212,132,110);
insert into RESULTATCGH values
(403,12,127,200,117,130);
insert into RESULTATCGH values
(402,12,128,210,123,115);
insert into RESULTATCGH values
(404,12,129,212,132,112);

insert into RESULTATCGH values
(101,3,34,220,127,130);
insert into RESULTATCGH values
(102,3,35,210,123,115);
insert into RESULTATCGH values
(100,3,36,212,132,132);
insert into RESULTATCGH values
(201,3,37,230,137,140);
insert into RESULTATCGH values
(302,3,38,205,133,115);
insert into RESULTATCGH values
(404,3,39,212,132,132);
insert into RESULTATCGH values
(403,3,310,230,137,140);
insert into RESULTATCGH values
(501,3,311,205,133,115);

insert into RESULTATCGH values
(100,8,1008,270,222,232);
insert into RESULTATCGH values
(401,8,4018,240,207,210);
insert into RESULTATCGH values
(101,8,1018,255,223,215);
insert into RESULTATCGH values
(102,8,1009,212,132,132);
insert into RESULTATCGH values
(201,8,4019,230,137,140);
insert into RESULTATCGH values
(301,8,5019,205,133,115);
insert into RESULTATCGH values
(302,8,10010,212,132,132);
insert into RESULTATCGH values
(402,8,40110,230,137,140);
insert into RESULTATCGH values
(403,8,30210,205,133,115);

insert into RESULTATCGH values
(404,8,10011,212,132,132);
insert into RESULTATCGH values
(501,8,40111,230,137,140);

insert into RESULTATCGH values
(301,17,30211,205,133,115);
insert into RESULTATCGH values
(402,17,10012,212,132,132);
insert into RESULTATCGH values
(401,17,40112,230,137,140);

insert into RESULTATCGH values
(102,19,10212,205,133,115);
insert into RESULTATCGH values
(201,19,10013,212,232,232);
insert into RESULTATCGH values
(301,19,40113,230,237,240);
insert into RESULTATCGH values
(302,19,10213,205,233,215);
insert into RESULTATCGH values
(403,19,10014,212,232,231);
insert into RESULTATCGH values
(401,19,40114,230,237,240);
insert into RESULTATCGH values
(402,19,10214,205,233,215);
insert into RESULTATCGH values
(404,19,10015,212,232,232);
insert into RESULTATCGH values
(100,19,40115,230,237,240);

insert into RESULTATCGH values
(201,11,10215,205,233,215);
insert into RESULTATCGH values
(100,11,10016,212,132,132);
insert into RESULTATCGH values
(101,11,10116,230,137,140);
insert into RESULTATCGH values
(102,11,10216,205,133,115);
insert into RESULTATCGH values
(301,11,10017,212,132,132);
insert into RESULTATCGH values
(302,11,10117,230,137,140);
insert into RESULTATCGH values
(402,11,40217,205,133,115);
insert into RESULTATCGH values
(401,11,10018,212,132,132);
insert into RESULTATCGH values
(404,11,10118,230,137,140);
insert into RESULTATCGH values
(403,11,40318,205,133,115);
insert into RESULTATCGH values
(501,11,10019,212,232,232);

```
/*  
insert into RESULTATCGH values  
(401,19,40119,230,237,240);  
insert into RESULTATCGH values  
(404,19,40419,205,233,215);  
  
insert into RESULTATCGH values  
(100,20,10020,212,232,232);  
insert into RESULTATCGH values  
(401,20,40120,230,237,240);  
insert into RESULTATCGH values  
(102,20,10220,205,233,215);  
*/  
commit;
```

```
insert into PARTICIPA_LACGH values  
(4,12345,100,'verde');  
insert into PARTICIPA_LACGH values  
(4,12345,101,'verde');  
insert into PARTICIPA_LACGH values  
(4,12345,102,'verde');  
insert into PARTICIPA_LACGH values  
(4,12345,201,'verde');  
insert into PARTICIPA_LACGH values  
(4,12345,301,'verde');  
insert into PARTICIPA_LACGH values  
(4,12345,302,'verde');  
insert into PARTICIPA_LACGH values  
(4,12345,401,'verde');  
insert into PARTICIPA_LACGH values  
(4,12345,402,'verde');  
insert into PARTICIPA_LACGH values  
(4,12345,403,'verde');  
insert into PARTICIPA_LACGH values  
(4,12345,404,'verde');  
insert into PARTICIPA_LACGH values  
(4,12345,501,'verde');
```

```
insert into PARTICIPA_LACGH values  
(14,45678,100,'maro');  
insert into PARTICIPA_LACGH values  
(14,45678,101,'maro');  
insert into PARTICIPA_LACGH values  
(14,45678,102,'maro');  
insert into PARTICIPA_LACGH values  
(14,45678,201,'maro');  
insert into PARTICIPA_LACGH values  
(14,45678,301,'maro');
```

insert into PARTICIPA_LACGH values
(14,45678,302,'maro');
insert into PARTICIPA_LACGH values
(14,45678,401,'maro');
insert into PARTICIPA_LACGH values
(14,45678,402,'maro');
insert into PARTICIPA_LACGH values
(14,45678,403,'maro');
insert into PARTICIPA_LACGH values
(14,45678,404,'maro');
insert into PARTICIPA_LACGH values
(14,45678,501,'maro');

insert into PARTICIPA_LACGH values
(15,45678,100,'maro');
insert into PARTICIPA_LACGH values
(15,45678,101,'maro');
insert into PARTICIPA_LACGH values
(15,45678,102,'maro');
insert into PARTICIPA_LACGH values
(15,45678,201,'maro');
insert into PARTICIPA_LACGH values
(15,45678,301,'maro');
insert into PARTICIPA_LACGH values
(15,45678,302,'maro');
insert into PARTICIPA_LACGH values
(15,45678,401,'maro');
insert into PARTICIPA_LACGH values
(15,45678,402,'maro');
insert into PARTICIPA_LACGH values
(15,45678,403,'maro');
insert into PARTICIPA_LACGH values
(15,45678,404,'maro');
insert into PARTICIPA_LACGH values
(15,45678,501,'maro');

insert into PARTICIPA_LACGH values
(16,34567,301,'portocaliu');
insert into PARTICIPA_LACGH values
(16,34567,401,'portocaliu');
insert into PARTICIPA_LACGH values
(16,34567,402,'portocaliu');
insert into PARTICIPA_LACGH values
(20,34567,301,'portocaliu');
insert into PARTICIPA_LACGH values
(20,34567,401,'portocaliu');
insert into PARTICIPA_LACGH values
(20,34567,402,'portocaliu');
insert into PARTICIPA_LACGH values
(21,34567,301,'portocaliu');
insert into PARTICIPA_LACGH values
(21,34567,401,'portocaliu');
insert into PARTICIPA_LACGH values

(21,34567,402,'portocaliu');

insert into REZULTATCGH values
(301,4,41,220,150,150);
insert into REZULTATCGH values
(401,4,42,210,143,147);
insert into REZULTATCGH values
(402,4,43,212,142,150);
insert into REZULTATCGH values
(101,4,44,220,127,130);
insert into REZULTATCGH values
(102,4,45,210,123,115);
insert into REZULTATCGH values
(100,4,46,212,132,132);
insert into REZULTATCGH values
(201,4,47,230,137,140);
insert into REZULTATCGH values
(302,4,48,205,133,115);
insert into REZULTATCGH values
(404,4,49,212,132,132);
insert into REZULTATCGH values
(403,4,410,230,137,140);
insert into REZULTATCGH values
(501,4,411,205,133,115);

insert into REZULTATCGH values
(201,14,141,205,233,215);
insert into REZULTATCGH values
(100,14,142,212,132,132);
insert into REZULTATCGH values
(101,14,143,230,137,140);
insert into REZULTATCGH values
(102,14,144,205,133,115);
insert into REZULTATCGH values
(301,14,145,212,132,132);
insert into REZULTATCGH values
(302,14,146,230,137,140);
insert into REZULTATCGH values
(402,14,147,205,133,115);
insert into REZULTATCGH values
(401,14,148,212,132,132);
insert into REZULTATCGH values
(404,14,149,230,137,140);
insert into REZULTATCGH values
(403,14,1410,205,133,115);
insert into REZULTATCGH values
(501,14,1411,212,232,232);

insert into REZULTATCGH values
(201,15,151,205,233,215);
insert into REZULTATCGH values
(100,15,152,212,132,132);

```

insert into RESULTATCGH values
(101,15,153,230,137,140);
insert into RESULTATCGH values
(102,15,154,205,133,115);
insert into RESULTATCGH values
(301,15,155,212,132,132);
insert into RESULTATCGH values
(302,15,156,230,137,140);
insert into RESULTATCGH values
(402,15,157,205,133,115);
insert into RESULTATCGH values
(401,15,158,212,132,132);
insert into RESULTATCGH values
(404,15,159,230,137,140);
insert into RESULTATCGH values
(403,15,1510,205,133,115);
insert into RESULTATCGH values
(501,15,1511,212,232,232);

```

```

insert into RESULTATCGH values
(301,16,161,205,133,115);
insert into RESULTATCGH values
(402,16,162,212,132,132);
insert into RESULTATCGH values
(401,16,163,230,137,140);
insert into RESULTATCGH values
(301,20,201,205,133,115);
insert into RESULTATCGH values
(402,20,202,212,132,132);
insert into RESULTATCGH values
(401,20,203,230,137,140);
insert into RESULTATCGH values
(301,21,211,205,133,115);
insert into RESULTATCGH values
(402,21,212,212,132,132);
insert into RESULTATCGH values
(401,21,213,230,137,140);

```

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

Cerinta:

Fiind dat id-ul unei competitii, vreau sa afisez toate etapele care fac parte din aceasta(trambulina, data,tipul si daca e etapa diurna sau nocturna). Vreau si sa adaug un an la varsta tuturor saritorilor care participa la ultima etapa din competitia respectiva.

-colectii utilizate: tablou_indexat, vector

```

CREATE OR REPLACE PROCEDURE punctul6_cgh(id_comp
COMPETITIECGH.id_competitie%TYPE)
IS

```

```

type r is record(
id_et etapacgh.id_etapa%type,
trambulina trambulinacgh.nume%type,
data_et etapacgh.data_etapa%type,
tip_et etapacgh.tip%type,
zi_et etapacgh.zi_noapte%type
);
TYPE tablou_indexat IS TABLE OF r INDEX BY BINARY_INTEGER;
t tablou_indexat;
ultima_etapa etapacgh.id_etapa%TYPE;
TYPE vector IS VARRAY(30) OF NUMBER(5);
v vector:= vector();
cnt number:=1;
var_id_saritor saritorcgh.id_saritor%type;

BEGIN

select distinct e.id_etapa, t.nume, e.data_etapa, e.tip,e.zi_noapte bulk collect into t
from etapacgh e,participa_lacgh p,trambulinacgh t
where p.id_competitie=id_comp and p.id_etapa=e.id_etapa and
t.id_trambulina=e.id_trambulina
order by e.data_etapa;

DBMS_OUTPUT.PUT_LINE('Competitia cu id-ul ' || id_comp || ' contine urmatoarele
etape: ');

FOR i IN t.FIRST..t.LAST LOOP
DBMS_OUTPUT.PUT_LINE('Etapa numarul ' || i || ' are loc pe ' || t(i).trambulina || ' in data
de ' || t(i).data_et);
if t(i).zi_et ='zi' then
DBMS_OUTPUT.PUT_LINE('Este o etapa diurna de ' || t(i).tip_et);
else
DBMS_OUTPUT.PUT_LINE('Este o etapa nocturna de ' || t(i).tip_et || '.');
end if;
DBMS_OUTPUT.PUT_LINE("");
END LOOP;

ultima_etapa:= t(t.LAST).id_et;
select distinct p.id_saritor bulk collect into v
from participa_lacgh p where p.id_etapa = ultima_etapa and p.id_competitie=id_comp;

FOR i IN v.FIRST..v.LAST LOOP
update SARITORCGH
set varsta=varsta+1
where id_saritor=v(i);
end loop;

END punctul6_cgh;

```

```
EXECUTE punctul6_cgh(34567);
--adauga doar la 401,402,301(ultima etapa e 21)
```

```
DROP PROCEDURE punctul6_cgh;
```

Tabelul de saritori inainte:

ID_SARITOR	NUME	PRENUME	GEN	VARSTA	GREUTATE	ID_ECHIPA
1	401	Kraft	Stefan	m	34	56 AUT
2	402	Horl	Jan	m	28	67 AUT
3	403	Huber	Daniel	m	28	64 AUT
4	404	Morgenstern	Thomas	m	43	66 AUT
5	100	Eisenbichler	Markus	m	32	56 GER
6	101	Geiger	Karl	m	26	67 GER
7	102	Wellinger	Andreas	m	28	64 GER
8	301	Kos	Lovro	m	28	67 SLO
9	302	Prevc	Peter	m	28	64 SLO
10	201	Tande	Daniel-Andre	m	26	67 NOV
11	901	Feldorean	Andrei	m	19	56 ROU

Tabelul de saritori dupa(varsta se mareste la saritorii 401, 402,301, care participa la ultima etapa din competitie):

ID_SARITOR	NUME	PRENUME	GEN	VARSTA	GREUTATE	ID_ECHIPA
1	401	Kraft	Stefan	m	35	56 AUT
2	402	Horl	Jan	m	29	67 AUT
3	403	Huber	Daniel	m	28	64 AUT
4	404	Morgenstern	Thomas	m	43	66 AUT
5	100	Eisenbichler	Markus	m	32	56 GER
6	101	Geiger	Karl	m	26	67 GER
7	102	Wellinger	Andreas	m	28	64 GER
8	301	Kos	Lovro	m	29	67 SLO
9	302	Prevc	Peter	m	28	64 SLO
10	201	Tande	Daniel-Andre	m	26	67 NOV
11	901	Feldorean	Andrei	m	19	56 ROU

Dbms Output

Buffer Size: 20000

Competitia cu id-ul 34567 contine urmatoarele etape:

Etapa numarul 1 are loc pe Heini-Klopfer-Skiflugschanze in data de 09-MAR-23
Este o etapa diurna de sarit

Etapa numarul 2 are loc pe Heini-Klopfer-Skiflugschanze in data de 10-MAR-23
Este o etapa diurna de sarit

Etapa numarul 3 are loc pe Vikersundbakken in data de 16-MAR-23
Este o etapa nocturna de sarit.

Etapa numarul 4 are loc pe Vikersundbakken in data de 17-MAR-23
Este o etapa nocturna de sarit.

Etapa numarul 5 are loc pe Vikersundbakken in data de 18-MAR-23
Este o etapa diurna de sarit

<pre>select id_saritor from participa_lacgh where id_competitie=34567 and id_etapa=21;</pre>	
Script Output x	Query Result x
SQL All Rows Fetched: 3 in 0.011 seconds	
ID_SARITOR	
1	301
2	401
3	402

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

Cerinta:

Obtineti primele doua echipa cu numarul cel mai mare de saritori. Afisati numele, numarul de saritori ale acestora, dar si numele, prenumele si varsta fiecarui saritor din respectivele echipe in parte.

-cursoare utilizate: ciclu cursor si cursor explicit parametrizat

```
CREATE OR REPLACE PROCEDURE punctul7_cgh
IS
--ciclu cursor
CURSOR c1 IS
SELECT e.id echipa, e.nume echipa, count(s.id_saritor) cnt
FROM echipacgh e, saritorcgh s
WHERE s.id echipa = e.id echipa
GROUP BY e.id echipa,e.nume echipa
order by count(s.id_saritor) desc;

CURSOR c2(parametru VARCHAR2) IS
SELECT nume,prenume,varsta
FROM saritorcgh
WHERE id echipa=parametru;
v_nume saritorcgh.nume%TYPE;
v_prenume saritorcgh.prenume%TYPE;
v_varsta saritorcgh.varsta%TYPE;
BEGIN
FOR i IN c1 LOOP
EXIT WHEN c1%ROWCOUNT>2 OR c1%NOTFOUND;
DBMS_OUTPUT.PUT_LINE('');
DBMS_OUTPUT.PUT_LINE('Echipa '||i.nume echipa||' are '||i.cnt||' saritori.');
```

```
DBMS_OUTPUT.PUT_LINE('Acestia sunt: ');
```

```

--cursor explicit
OPEN c2(i.id echipa);
LOOP
    FETCH c2 INTO v_nume,v_prename,v_varsta;
    EXIT WHEN c2%NOTFOUND;
    DBMS_OUTPUT.PUT_LINE('$'|| v_nume||' '|| v_prename||' cu varsta de '||v_varsta);
END LOOP;
CLOSE c2;
END LOOP;
END punctul7_cgh;
/
EXECUTE punctul7_cgh;
DROP PROCEDURE punctul7_cgh;



```

Rezultat:

```



END LOOP;
END punctul7_cgh;
/
EXECUTE punctul7_cgh;
DROP PROCEDURE punctul7_cgh;

```

Script Output x Query Result x
  | Task completed in 0.065 seconds

PL/SQL procedure successfully completed.

Dbms Output

  | Buffer Size: 20000

Echipa Austria are 4 saritori.

Acestia sunt:

\$Kraft Stefan cu varsta de 35

\$Horl Jan cu varsta de 29

\$Huber Daniel cu varsta de 28

\$Morgenstern Thomas cu varsta de 43

Echipa Germania are 3 saritori.

Acestia sunt:

\$Eisenbichler Markus cu varsta de 32

\$Geiger Karl cu varsta de 26

\$Wellinger Andreas cu varsta de 28

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. Apelați subprogramul astfel încât să evidențiați toate cazurile tratat.

Cerinta:

Se da numele unei tari. Afisati data etapei care are loc in respectiva tara doar daca aceasta este unica.

-exceptii: no_data_found, too_many_rows

-tabele utilizate: orascgh, trambulinacgh, etapacgh

--Am inserat urmatoarele date in tabele pentru a exemplifica cerinta:

```
insert into TARACGH values
```

```
(9,'Romania');
```

```
insert into ORASCGH values
```

```
(91,9,'Rasnov');
```

```
insert into TRAMBULINACGH values
```

```
(911,91,'Valea Carbunarii',90,100,105);
```

```
insert into ETAPACGH values
```

```
(911,99,'18-FEB-2023','sarit','zi');
```

```
CREATE OR REPLACE FUNCTION punctul8_cgh
```

```
(v_nume_tara taracgh.ume%type DEFAULT 'Germania')
```

```
RETURN etapacgh.data_etapa%type IS
```

```
    v_data etapacgh.data_etapa%type;
```

```
    v_id_tara taracgh.id_tara%type;
```

```
BEGIN
```

```
    select id_tara into v_id_tara
```

```
    from taracgh where lower(v_nume_tara)=lower(ume);
```

```
    select e.data_etapa into v_data
```

```
    from orascgh o JOIN trambulinacgh t ON(t.id_oras=o.id_oras)
```

```
    JOIN etapacgh e ON(e.id_trambulina=t.id_trambulina)
```

```
    where o.id_tara=v_id_tara;
```

```
    DBMS_OUTPUT.PUT_LINE('Data etapei din ' || v_nume_tara || ' este ');
```

```

    RETURN v_data;

EXCEPTION

    WHEN NO_DATA_FOUND THEN

        DBMS_OUTPUT.PUT_LINE('Nu exista tari cu numele ' || v_nume_tara || '.');

        RETURN null;

    WHEN TOO_MANY_ROWS THEN

        DBMS_OUTPUT.PUT_LINE('Exista mai mult de o etapa in ' || v_nume_tara || '.');

        RETURN null;

    WHEN OTHERS THEN

        DBMS_OUTPUT.PUT_LINE('Alta eroare!');

        RETURN null;

END punctul8_cgh;

/

BEGIN

--too_many_rows

DBMS_OUTPUT.PUT_LINE(punctul8_cgh);

END;/

BEGIN

--too_many_rows

DBMS_OUTPUT.PUT_LINE(punctul8_cgh('Norvegia'));

END;/

--merge

BEGIN

DBMS_OUTPUT.PUT_LINE( punctul8_cgh('Romania') || '.');

END;/

BEGIN

--no_data_found

DBMS_OUTPUT.PUT_LINE(punctul8_cgh('Irlanda'));

END;/

drop function punctul8_cgh;

```


Rezultatele la cele 4 apeluri:

```
Exista mai mult de o etapa in Germania.
```

```
Exista mai mult de o etapa in Norvegia.
```

```
Data etapei din Romania este  
18-FEB-23.
```

```
Nu exista tari cu numele Irlanda.
```

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.

Cerinta:

Se da numele unei echipe si antrenorul acesteia. Pentru fiecare saritor din aceasta echipa care a suferit candva o accidentare vreau sa afisez numele, prenumele si procentul de etape de zi din numarul de etape de zi sau de noapte la care a participat si a luat mai mult de 100 de puncte.

-exceptii: no_data_found, too_many_rows, value_error, zero_divide

-tabele utilizate: etapacgh, echipacgh, saritorcgh, rezultatcgh, accidentarecgh, sponsorizeazacgh, sponsorcgh

Am inserat urmatoarele date in tabele pentru a exemplifica cerinta:

```
insert into SARITORCGH values
```

```
('901','Feldorean','Andrei','m',19,56,'ROU');
```

```
insert into ACCIDENTARECGH values
```

```
(999,901,'05-DEC-2022');
```

```
insert into ECHIPACGH values
```

```
('FLu','flas','Abi Lanon',4300);
```

```
insert into ECHIPACGH values
```

```
('FLd','flas','Abi Lanon',4300);  
insert into ECHIPACGH values  
('FLo','flas3','Abi Lanon',4300);
```

```
CREATE OR REPLACE PROCEDURE punctul9_cgh  
(v_num_echipa echipacgh.ume_echipa%type,  
v_ume_antrenor echipacgh.antrenor%type) IS
```

```
type r is record(  
ume saritorcgh.ume%type,  
preume saritorcgh.preume%type,  
cnt1 number,  
cnt2 number  
);
```

```
v_id_echipa echipacgh.id_echipa%type;
```

```
TYPE tablou_indexat IS TABLE OF r INDEX BY BINARY_INTEGER;
```

```
t tablou_indexat;
```

```
cnt2 number;
```

```
procent float:=0;
```

```
BEGIN
```

```
select id_echipa into v_id_echipa
```

```
from echipacgh where lower(v_ume_echipa)=lower(ume_echipa);
```

```
select distinct s.ume,s.preume, nvl((select count(et.id_etapa) from rezultacgh r,  
etapacgh et
```

```
where r.id_saritor=s.id_saritor and r.puncte>100
```

```
and et.id_etapa=r.id_etapa and et.tip='sarit'),0) cnt1,
```

```
nvl((select count(r.id_etapa) from rezultacgh r
```

```
where r.id_saritor=s.id_saritor and r.puncte>100),0) cnt2 BULK COLLECT  
INTO T
```

```
from echipacgh e JOIN saritorcgh s ON (s.id_echipa=e.id_echipa)
```

```
JOIN accidentarecgh a ON (a.id_saritor=s.id_saritor)
```

```
JOIN sponsorizezacgh sp ON (sp.id_echipa=e.id_echipa)
```

```

JOIN sponsorcgh spo ON (spo.id_sponsor=sp.id_sponsor)

where e.id echipa=v_id echipa and e.antrenor=v_num_e_antrenor

and spo.denumire='adidas';


DBMS_OUTPUT.PUT_LINE('Echipa cu numele ' || v_num_e echipa || ' contine urmatorii
saritori care au avut accidentari: ');


FOR i IN t.FIRST..t.LAST LOOP

    procent:=round(t(i).cnt1/t(i).cnt2,2)*100;

    DBMS_OUTPUT.PUT_LINE(t(i).nume || ' ' || t(i).prenume || ', care are raportul de
participare la etapele de sarit = ' || procent || '%');

END LOOP;

EXCEPTION

    WHEN NO_DATA_FOUND THEN

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


    WHEN TOO_MANY_ROWS THEN

        DBMS_OUTPUT.PUT_LINE('Exista mai multe echipe cu numele ' || v_num_e echipa ||
'.');


    WHEN ZERO_DIVIDE THEN

        DBMS_OUTPUT.PUT_LINE('Eroare: Exista un saritor in echipa ' || v_num_e echipa || ' care
nu a participat la nicio etapa.');
```

```

    WHEN VALUE_ERROR THEN

        DBMS_OUTPUT.PUT_LINE('Eroare: Ati gresit numele antrenorului sau adidas nu este
sponsor.');
```

```

    WHEN OTHERS THEN

        DBMS_OUTPUT.PUT_LINE('Alta eroare!');


END punctul9_cgh;

/

--zero_divide

EXECUTE punctul9_cgh('Romania','Dumitru Petre');
```

```

--merge
EXECUTE punctul9_cgh('Slovenia','Abi Lanon');

--no_data_found
EXECUTE punctul9_cgh('Islanda','Abi Lanon');

--too_many_rows
EXECUTE punctul9_cgh('flas','Abi Lanon');

--value_error(numere antrenor gresit)
EXECUTE punctul9_cgh('Slovenia','Abiel Lanov');

--value_error(echipa nu il are pe adidas drept sponsor)
EXECUTE punctul9_cgh('flas3','Abi Lanon');

drop procedure punctul9_cgh;

```

Echipa cu numele Romania contine urmatorii saritori care au avut accidentari:
 Eroare: Exista un saritor in echipa Romania care nu a participat la nicio etapa.

Echipa cu numele Slovenia contine urmatorii saritori care au avut accidentari:
 Kos Lovro, care are raportul de participare la etapele de sarit = 83%
 Prevc Peter, care are raportul de participare la etapele de sarit = 75%

Nu exista echipe cu numele Islanda.

Exista mai multe echipe cu numele flas.

Echipa cu numele Slovenia contine urmatorii saritori care au avut accidentari:
 Eroare: Ati gresit numele antrenorului sau adidas nu este sponsor.

Echipa cu numele flas3 contine urmatorii saritori care au avut accidentari:
 Eroare: Ati gresit numele antrenorului sau adidas nu este sponsor.

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

Cerinta:

Vreau un trigger care sa nu ma lase sa inserez in tabelul saritorcgh daca am deja mai mult de 10 saritori de gen masculin in tabel.

```

CREATE OR REPLACE TRIGGER trig1_cgh
BEFORE INSERT ON saritorcgh
DECLARE
cnt number:=0;
BEGIN
select count(*)

```

```

into cnt
from saritorcgh
where gen='m';
IF (cnt>10) THEN
RAISE_APPLICATION_ERROR(-20001,'Sunt deja prea multi saritori inscisi.');
```

END IF;

END;

/

```

DROP TRIGGER trig1_cgh;

insert into SARITORCGH values
('902','Cacina','Andrei','m',18,56,'ROU');
```

insert into SARITORCGH values

```

('903','Haralambie','Dana','f',23,60,'ROU');
```

```

--eroare
insert into SARITORCGH values
('902','Cacina','Andrei','m',18,56,'ROU');
```

--eroare

```

insert into SARITORCGH values
('903','Haralambie','Dana','f',23,60,'ROU');
```

Script Output x Query Result x

Task completed in 0.285 seconds

Error starting at line : 272 in command -

```

insert into SARITORCGH values
('902','Cacina','Andrei','m',18,56,'ROU')
```

Error report -

ORA-20001: Sunt deja prea multi saritori inscisi.

ORA-06512: at "GRUPA231.TRIG1_CGH", line 9

ORA-04088: error during execution of trigger 'GRUPA231.TRIG1_CGH'

Error starting at line : 275 in command -

```

insert into SARITORCGH values
('903','Haralambie','Dana','f',23,60,'ROU')
```

Error report -

ORA-20001: Sunt deja prea multi saritori inscisi.

ORA-06512: at "GRUPA231.TRIG1_CGH", line 9

ORA-04088: error during execution of trigger 'GRUPA231.TRIG1_CGH'

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

Cerinta:

Realizez un trigger care te lasa sa modifici recordul unei trambuline doar daca noua valoare este mai mare decat cea veche.

```
CREATE OR REPLACE TRIGGER trig2_cgh
BEFORE UPDATE OF record ON trambulinacgh
FOR EACH ROW
BEGIN
IF (:NEW.record < :OLD.record) THEN
RAISE_APPLICATION_ERROR(-20002,'Recordul nou trebuie sa fie mai mare decat cel anterior.');
```

END IF;

END;

/


```
UPDATE trambulinacgh
SET record=record-1
WHERE id_trambulina in(212,321,411);
```



```
DROP TRIGGER trig2_cgh;
```

```
UPDATE trambulinacgh
SET record=record-1
WHERE id_trambulina in(212,321,411);
```

```
DROP TRIGGER trig2_cgh;
drop table actiunicgh;
```

Script Output x Query Result x
Task completed in 0.273 seconds

```
Error starting at line : 290 in command -
UPDATE trambulinacgh
SET record=record-1
WHERE id_trambulina in(212,321,411)
Error report -
ORA-20002: Recordul nou trebuie sa fie mai mare decat cel anterior.
ORA-06512: at "GRUPA231.TRIG2_CGH", line 3
ORA-04088: error during execution of trigger 'GRUPA231.TRIG2_CGH'
```

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

Cerinta:

Realizez un trigger care atunci cand realizez o actiune de stergere, creare sau modificare sa imi memoreze actiunea(tipul actiunii, tipul obiectului, numele obiectului si data) intr-un tabel numit actiunecgh1.

```
CREATE TABLE actiunicgh1
```

```
(actiune varchar(40),
```

```
tip_obiect varchar(40),
```

```
nume_obiect varchar(40),
```

```
data date
```

```
);
```

```
CREATE OR REPLACE TRIGGER trig3_cgh
```

```
AFTER CREATE OR DROP OR ALTER ON SCHEMA
```

```
BEGIN
```

```
INSERT INTO actiunicgh1
```

```
VALUES (SYS.SYSEVENT,SYS.DICTIONARY_OBJ_TYPE,SYS.DICTIONARY_OBJ_NAME, SYSTIMESTAMP);
```

```
END;
```

```
/
```

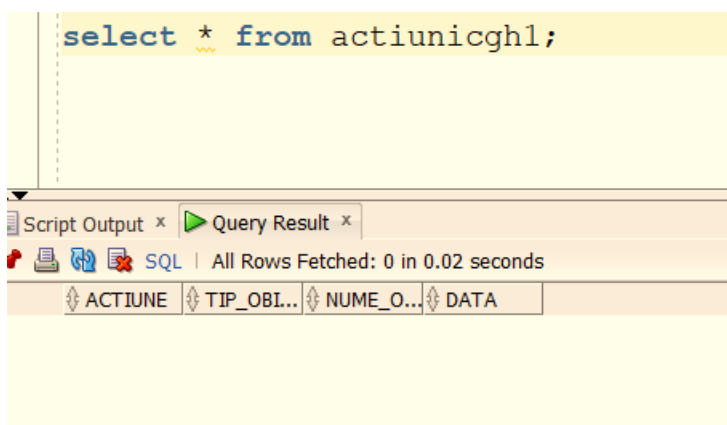
```
drop table test;
```

```
create table test(test varchar(20));
```

```
select * from actiunicgh1;
```

```
DROP TRIGGER trig3_cgh;
```

Inainte:



Dupa:

```
drop table test;
create table test(test varchar(20));
select * from actiunicgh1;
```

cript Output x Query Result x

SQL | All Rows Fetched: 2 in 0.01 seconds

	ACTIUNE	TIP_OBJECT	NUME_OBJECT	DATA
1	DROP	TABLE	TEST	12-JAN-23
2	CREATE TABLE	TEST		12-JAN-23

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

```
CREATE OR REPLACE PACKAGE pachet_cgh AS
```

```
  PROCEDURE punctul6_cgh(id_comp COMPETITIECGH.id_competitie%TYPE);
```

```
  PROCEDURE punctul7_cgh;
```

```
  FUNCTION punctul8_cgh
```

```
    (v_ume_tara taracgh.ume%type DEFAULT 'Germania')
```

```
  RETURN etapacgh.data_etapa%type;
```

```
  PROCEDURE punctul9_cgh
```

```
    (v_ume echipa echipacgh.ume_echipa%type,
```

```
    v_ume_antrenor echipacgh.antrenor%type);
```

```
END pachet_cgh;
```

```
CREATE OR REPLACE PACKAGE BODY pachet_cgh AS
```

```
  PROCEDURE punctul6_cgh(id_comp COMPETITIECGH.id_competitie%TYPE)
```

```
  IS
```

```
    type r is record(
```

```
      id_et etapacgh.id_etapa%type,
```

```
      trambulina trambulinacgh.ume%type,
```

```
      data_et etapacgh.data_etapa%type,
```

```
      tip_et etapacgh.tip%type,
```

```
      zi_et etapacgh.zi_noapte%type
```

```
    );
```

```
  TYPE tablou_indexat IS TABLE OF r INDEX BY BINARY_INTEGER;
```



```
t tablou_indexat;
```

```
ultima_etapa etapacgh.id_etapa%TYPE;
```

```
TYPE vector IS VARRAY(30) OF NUMBER(5);
```

```
v vector:= vector();
```

```
cnt number:=1;
```

```
var_id_saritor saritorcgh.id_saritor%type;
```

```
BEGIN
```

```
select distinct e.id_etapa, t.nume, e.data_etapa, e.tip,e.zi_noapte bulk collect into t
```

```
from etapacgh e,participa_lacgh p,trambulinacgh t
```

```
where p.id_competitie=id_comp and p.id_etapa=e.id_etapa and t.id_trambulina=e.id_trambulina
```

```
order by e.data_etapa;
```

```
DBMS_OUTPUT.PUT_LINE('Competitia cu id-ul ' || id_comp || ' contine urmatoarele etape: ');
```

```
FOR i IN t.FIRST..t.LAST LOOP
```

```
DBMS_OUTPUT.PUT_LINE('Etapa numarul ' || i || ' are loc pe ' || t(i).trambulina || ' in data de ' || t(i).data_et);
```

```
if t(i).zi_et ='zi' then
```

```
DBMS_OUTPUT.PUT_LINE('Este o etapa diurna de ' || t(i).tip_et);
```

```
else
```

```
DBMS_OUTPUT.PUT_LINE('Este o etapa nocturna de ' || t(i).tip_et || '.');
```

```
end if;
```

```
DBMS_OUTPUT.PUT_LINE("");
```

```
END LOOP;
```

```
ultima_etapa:= t(t.LAST).id_et;
```

```
select distinct p.id_saritor bulk collect into v
```

```
from participa_lacgh p where p.id_etapa = ultima_etapa and p.id_competitie=id_comp;
```

```
--un saritor poate concura la o anumita etapa fara sa fie in toate competitiiile din care face parte acea etapa
```

```
FOR i IN v.FIRST..v.LAST LOOP
```

```
update SARITORCGH
```

```
set varsta=varsta+1
```

```
where id_saritor=v(i);
```

```
end loop;
```

```
END punctul6_cgh;
```

```
PROCEDURE punctul7_cgh
```

```
IS
```

```
CURSOR c1 IS
```

```
SELECT e.id echipa, e.num_echipa, count(s.id_saritor) cnt
```

```
FROM echipacgh e, saritorcgh s
```

```
WHERE s.id_echipa = e.id_echipa
```

```
GROUP BY e.id_echipa,e.num_echipa
```

```
order by count(s.id_saritor) desc;
```

```
CURSOR c2(parametru VARCHAR2) IS
```

```
SELECT nume,prenume,varsta
```

```
FROM saritorcgh
```

```
WHERE id_echipa=parametru;
```

```
v_num saritorcgh.num%TYPE;
```

```
v_prenume saritorcgh.prenume%TYPE;
```

```
v_varsta saritorcgh.varsta%TYPE;
```

```
BEGIN
```

```
FOR i IN c1 LOOP
```

```
EXIT WHEN c1%ROWCOUNT>2 OR c1%NOTFOUND;
```

```
DBMS_OUTPUT.PUT_LINE("");
```

```
DBMS_OUTPUT.PUT_LINE('Echipa ' || i.num_echipa || ' are ' || i.cnt || ' saritori.');
```

```
DBMS_OUTPUT.PUT_LINE('Acestia sunt: ');
```

```
--cursor explicit
```

```
OPEN c2(i.id echipa);
```

```
LOOP
```

```
FETCH c2 INTO v_num, v_prenume, v_varsta;
```

```
EXIT WHEN c2%NOTFOUND;
```

```
DBMS_OUTPUT.PUT_LINE('§' || v_num || ' ' || v_prenume || ' cu varsta de ' || v_varsta);
```

```
END LOOP;
```

```
CLOSE c2;
```

```
END LOOP;
```

```
END punctul7_cgh;
```

```
FUNCTION punctul8_cgh
```

```
(v_num_tara taracgh.nume%type DEFAULT 'Germania')
```

```
RETURN etapacgh.data_etapa%type IS
```

```
v_data etapacgh.data_etapa%type;
```

```
v_id_tara taracgh.id_tara%type;
```

```
BEGIN
```

```
select id_tara into v_id_tara
```

```
from taracgh where lower(v_num_tara)=lower(nume);
```

```
select e.data_etapa into v_data
```

```
from orascgh o JOIN trambulinacgh t ON(t.id_oras=o.id_oras)
```

```
JOIN etapacgh e ON(e.id_trambulina=t.id_trambulina)
```

```
where o.id_tara=v_id_tara;
```

```
DBMS_OUTPUT.PUT_LINE('Data etapei din ' || v_num_tara || ' este ');
```

```
RETURN v_data;
```

```
EXCEPTION
```

```
WHEN NO_DATA_FOUND THEN
```

```
DBMS_OUTPUT.PUT_LINE('Nu exista tari cu numele ' || v_num_tara || '.');
```

```
RETURN null;
```

```
WHEN TOO_MANY_ROWS THEN
```

```
DBMS_OUTPUT.PUT_LINE('Exista mai mult de o etapa in ' || v_num_tara || '.');
```

```
RETURN null;
```

```
WHEN OTHERS THEN
```

```
DBMS_OUTPUT.PUT_LINE('Alta eroare!');
```

```
RETURN null;
```

```
END punctul8_cgh;
```

```
PROCEDURE punctul9_cgh
```

```
(v_ume_echipa echipacgh.ume_echipa%type,
```

```
v_ume_antrenor echipacgh.antrenor%type) IS
```

```
type r is record(
```

```
ume saritorcgh.ume%type,
```

```
preume saritorcgh.preume%type,
```

```
cnt1 number,
```

```
cnt2 number
```

```
);
```

```
v_id_echipa echipacgh.id_echipa%type;
```

```
TYPE tablou_indexat IS TABLE OF r INDEX BY BINARY_INTEGER;
```

```
t tablou_indexat;
```

```
cnt2 number;
```

```
procent float:=0;
```

```
BEGIN
```

```
select id_echipa into v_id_echipa
```

```
from echipacgh where lower(v_ume_echipa)=lower(ume_echipa);
```

```
select distinct s.ume,s.preume, nvl((select count(et.id_etapa) from rezultatcgh r, etapacgh et
```

```
where r.id_saritor=s.id_saritor and r.punte>100
```

```
and et.id_etapa=r.id_etapa and et.tip='sarit'),0) cnt1,
```

```
nvl((select count(r.id_etapa) from rezultatcgh r
```

```
where r.id_saritor=s.id_saritor and r.punte>100),0) cnt2 bulk collect into t
```

```
from echipacgh e JOIN saritorcgh s ON (s.id_echipa=e.id_echipa)
```

```
JOIN accidentarecgh a ON (a.id_saritor=s.id_saritor)
```

```
JOIN sponsorizezacgh sp ON (sp.id echipa=e.id echipa)
```

```
JOIN sponsorcgh spo ON (spo.id_sponsor=sp.id_sponsor)
```

```
where e.id echipa=v_id echipa and e.antrenor=v_ume_antrenor
```

```
and spo.denumire='adidas';
```

```
DBMS_OUTPUT.PUT_LINE('Echipa cu numele ' || v_ume_echipa || ' contine urmatorii saritori care au  
avut accidentari: ');
```

```
FOR i IN t.FIRST..t.LAST LOOP
```

```
procent:=round(t(i).cnt1/t(i).cnt2,2)*100;
```

```
DBMS_OUTPUT.PUT_LINE(t(i).nume || ' ' || t(i).prenume || ', care are raportul de participare la etapele  
de sarit = ' || procent || '%');
```

```
END LOOP;
```

```
EXCEPTION
```

```
WHEN NO_DATA_FOUND THEN
```

```
DBMS_OUTPUT.PUT_LINE('Nu exista echipe cu numele ' || v_ume_echipa || '.');
```

```
WHEN TOO_MANY_ROWS THEN
```

```
DBMS_OUTPUT.PUT_LINE('Exista mai multe echipe cu numele ' || v_ume_echipa || '.');
```

```
WHEN ZERO_DIVIDE THEN
```

```
DBMS_OUTPUT.PUT_LINE('Eroare: Exista un saritor in echipa ' || v_ume_echipa || ' care nu a  
participat la nicio etapa.');
```

```
WHEN VALUE_ERROR THEN
```

```
DBMS_OUTPUT.PUT_LINE('Eroare: Ati gresit numele antrenorului sau adidas nu este sponsor.');
```

```
WHEN OTHERS THEN
```

```
DBMS_OUTPUT.PUT_LINE('Alta eroare!');
```

```
END punctul9_cgh;
```

```
END pachet_cgh;
```

```

DROP PACKAGE pachet_cgh;

BEGIN

  DBMS_OUTPUT.PUT_LINE('6:');
  pachet_cgh.punctul6_cgh(34567);
  DBMS_OUTPUT.PUT_LINE('7:');
  pachet_cgh.punctul7_cgh;
  DBMS_OUTPUT.PUT_LINE("");
  DBMS_OUTPUT.PUT_LINE('8:');
  DBMS_OUTPUT.PUT_LINE( pachet_cgh.punctul8_cgh('Romania') || '.');
  DBMS_OUTPUT.PUT_LINE("");
  DBMS_OUTPUT.PUT_LINE('9:');
  pachet_cgh.punctul9_cgh('Slovenia','Abi Lanon');

END;

```

```
Package PACHET_CGH compiled
```

```
Package Body PACHET_CGH compiled
```

Rezultat:

```

6:
Competitia cu id-ul 34567 contine urmatoarele etape:
Etapa numarul 1 are loc pe Heini-Klopfer-Skiflugschanze in data de 09-MAR-23
Este o etapa diurna de sarit

Etapa numarul 2 are loc pe Heini-Klopfer-Skiflugschanze in data de 10-MAR-23
Este o etapa diurna de sarit

Etapa numarul 3 are loc pe Vikersundbakken in data de 16-MAR-23
Este o etapa nocturna de sarit.

Etapa numarul 4 are loc pe Vikersundbakken in data de 17-MAR-23
Este o etapa nocturna de sarit.

Etapa numarul 5 are loc pe Vikersundbakken in data de 18-MAR-23
Este o etapa diurna de sarit

7:

Echipa Austria are 4 saritori.
Acestia sunt:
$Kraft Stefan cu varsta de 39
$Horl Jan cu varsta de 33
$Huber Daniel cu varsta de 28
$Morgenstern Thomas cu varsta de 43

Echipa Germania are 3 saritori

```

Echipa Germania are 3 saritori.

Acestia sunt:

\$Eisenbichler Markus cu varsta de 32

\$Geiger Karl cu varsta de 26

\$Wellinger Andreas cu varsta de 28

8:

Data etapei din Romania este

18-FEB-23.

9:

Echipa cu numele Slovenia contine urmasorii saritori care au avut accidentari:

Kos Lovro, care are raportul de participare la etapele de sarit = 83%

Prevc Peter, care are raportul de participare la etapele de sarit = 75%

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

Cerinta:

Vreau sa afisez competitiiile din care face parte prima etapa la care participa saritorul cu cel mai mare punctaj total adunat la etapele din competitia care are cele mai multe etape, dar si top 3 saritori de aici.

Explicatie:

- in functia 2 calculez competitia cu cele mai multe etape, o afisez si o trimit ca parametru procedurii1
- in procedura 1 aflu si afisez top trei saritori ordonati descrescator dupa punctajul total la competitia transmisa ca parametru (suma punctajelor din etapele care fac parte din competitia respectiva) si il trimit pe primul drept parametru pentru functia 3
- in functia 3 aflu prima etapa la care a participat saritorul dat ca parametru, o afisez si o trimit drept parametru in procedura 4
- in procedura 4 afisez toate competitiiile(nume, data inceput, data final) din care face parte etapa trimisa ca parametru.

-tipuri de date utilizate: tablou indexat, ciclu cursor cu subcereri, ciclu cursor

create or replace PACKAGE pachet14_cgh AS

--procedura 1

PROCEDURE p1_cgh(id_comp COMPETITIECGH.id_competitie%TYPE);

FUNCTION p2_cgh

RETURN participa_lacgh.id_competitie%type;

FUNCTION p3_cgh(id_sar saritorcgh.id_saritor%type)

RETURN etapacgh.id_etapa%type;

PROCEDURE p4_cgh(id_et etapacgh.id_etapa%type);

```

END pachet14_cgh;

create or replace PACKAGE BODY pachet14_cgh AS

  PROCEDURE p1_cgh(id_comp COMPETITIECGH.id_competitie%TYPE)
  IS
    type r is record(
      id_sar saritorcgh.id_saritor%type,
      nume saritorcgh.nume%type,
      prenume saritorcgh.prenume%type,
      puncte number
    );

    --tablou indexat
    TYPE tablou_indexat IS TABLE OF r INDEX BY BINARY_INTEGER;

    t tablou_indexat;
    n varchar(30);

  BEGIN

    select nume into n
      from competitiecgh
     where id_competitie=id_comp;

    select s.id_saritor, s.nume, s.prenume, sum(r.puncte) bulk collect into t
      from participa_lacgh p, rezultatcgh r, saritorcgh s
     where p.id_competitie=id_comp and p.id_saritor=s.id_saritor
        and s.id_saritor=r.id_saritor and p.id_etapa=r.id_etapa
     group by s.id_saritor, s.nume, s.prenume
     order by sum(r.puncte) desc;

    FOR i IN 1..3 LOOP

      DBMS_OUTPUT.PUT_LINE("");

      DBMS_OUTPUT.PUT_LINE('Saritorul ' || t(i).nume || ' ' || t(i).prenume || ' este pe podium in competitia ' || n);

      --ciclu cursor cu subcereri

      FOR j IN (select distinct ss.id echipa ec, ss.id_sponsor sp
        from echipacgh e, saritorcgh s, sponsorizeazacgh ss

```



```
where s.id echipa=e.id echipa and e.id echipa=ss.id echipa
```

```
and t(i).id_sar=s.id_saritor
```

```
) LOOP
```

```
update sponsorizeazacgh
```

```
set suma=suma+100
```

```
where id echipa=j.ec and id_sponsor=j.sp;
```

```
END LOOP;
```

```
END LOOP;
```

```
DBMS_OUTPUT.PUT_LINE("");
```

```
--apelez pt saritorul de pe primul loc
```

```
--vr sa obtin id-ul primei etape la care participa
```

```
p4_cgh(p3_cgh(t(1).id_sar));
```

```
END p1_cgh;
```

```
--functia 2
```

```
FUNCTION p2_cgh
```

```
RETURN participa_lacgh.id_competitie%type IS
```

```
v_id participa_lacgh.id_competitie%type;
```

```
nr_max number;
```

```
n varchar(30);
```

```
BEGIN
```

```
select max(cnt) into nr_max
```

```
from (select count(id_etapa) as cnt
```

```
from (select distinct id_competitie, id_etapa from participa_lacgh)
```

```
group by id_competitie);
```

```
select distinct p.id_competitie into v_id
```

```
from participa_lacgh p
```

```
where nr_max=(select count(id_etapa)
```

```
from (select distinct id_competitie, id_etapa from participa_lacgh)
```

```
where id_competitie=p.id_competitie);
```

```
select nume into n
```

```
from competitiecgh
```

```
where id_competitie=v_id;
```

```
DBMS_OUTPUT.PUT_LINE('Competitia cu cele mai multe etape este ' || n);
```

```
RETURN v_id;
```

```
END p2_cgh;
```

--functia 3

```
FUNCTION p3_cgh(id_sar saritorcgh.id_saritor%type)
```

```
RETURN etapacgh.id_etapa%type IS
```

```
TYPE tablou_indexat IS TABLE OF etapacgh.id_etapa%type INDEX BY BINARY_INTEGER;
```

```
t tablou_indexat;
```

```
num varchar(20);
```

```
prenum varchar(20);
```

```
BEGIN
```

```
select e.id_etapa bulk collect into t
```

```
from rezultatcgh r, etapacgh e
```

```
where r.id_saritor=id_sar and e.id_etapa=r.id_etapa
```

```
order by e.data_etapa;
```

```
select s.num,s.prenume
```

```
into num,prenum
```

```
from saritorcgh s
```

```
where s.id_saritor=id_sar;
```

```
DBMS_OUTPUT.PUT_LINE('Prima etapa la care participa ' || num || ' ' || prenum || ' are id-ul ' || t(1));
```

```
DBMS_OUTPUT.PUT_LINE("");
```

```
RETURN t(1);
```

```
END p3_cgh;
```

--procedura 4

```
PROCEDURE p4_cgh(id_et etapacgh.id_etapa%type)
```

```
IS
```

```
--ciclu cursor
```

```
CURSOR c IS select distinct c.num nume, c.data_inceput d1, c.data_final d2
```

```
from participa_lacgh p, competitiecgh c
```

```
where p.id_competitie=c.id_competitie
```

```
and p.id_etapa=id_et;
```

```
BEGIN
```

```
DBMS_OUTPUT.PUT_LINE('Etapa cu id-ul ' || id_et || ' face parte din competitile:');
```

```
FOR i in c LOOP
```

```
DBMS_OUTPUT.PUT_LINE(i.nume || ' care incepe pe ' || i.d1 || ' si se termina pe ' || i.d2);
```

```
END LOOP;
```

```
END p4_cgh;
```

```
END pachet14_cgh;
```

```
--Apel
```

```
BEGIN
```

```
pachet14_cgh.p1_cgh(pachet14_cgh.p2_cgh);
```

```
END;
```

The screenshot shows a database IDE with two main panes. The top pane, titled 'Worksheet' and 'Query Builder', contains a PL/SQL script. The script starts with 'BEGIN', followed by an indented call to 'pachet14_cgh.p1_cgh(pachet14_cgh.p2_cgh);', and ends with 'END;'. The bottom pane is divided into two sections. The top section, 'Script Output', shows the execution results: 'Package PACHET14_CGH compiled', 'Package Body PACHET14_CGH compiled', and 'PL/SQL procedure successfully completed.' The bottom section, 'Dbms Output', shows the output of the PL/SQL procedure, which lists the stages of the 'Mondial' competition and the participants on the podium for each stage.

```
BEGIN
    pachet14_cgh.p1_cgh(pachet14_cgh.p2_cgh);
END;
```

Script Output x Query Result x
Task completed in 0.118 seconds

Package PACHET14_CGH compiled

Package Body PACHET14_CGH compiled

PL/SQL procedure successfully completed.

Dbms Output
Buffer Size: 20000

Competitia cu cele mai multe etape este Mondial

Saritorul Kraft Stefan este pe podium in competitia Mondial

Saritorul Kos Lovro este pe podium in competitia Mondial

Saritorul Horl Jan este pe podium in competitia Mondial

Prima etapa la care participa Kraft Stefan are id-ul 3

Etapa cu id-ul 3 face parte din competitile:

Zbor cu schiurile care incepe pe 02-DEC-22 si se termina pe 05-DEC-23

Mondial care incepe pe 21-NOV-22 si se termina pe 18-MAR-23