Proiect SGBD

Anul 2- informatica

Gestionarea Campionatelor de tenis de masa din Bucuresti

Dilirici Mihai – grupa 243

1. *Descrierea bazei de date si a utilitatii acesteia.*

Baza de date se ocupa de gestionarea capionatelor de tenis de masa din orasul Bucuresti.

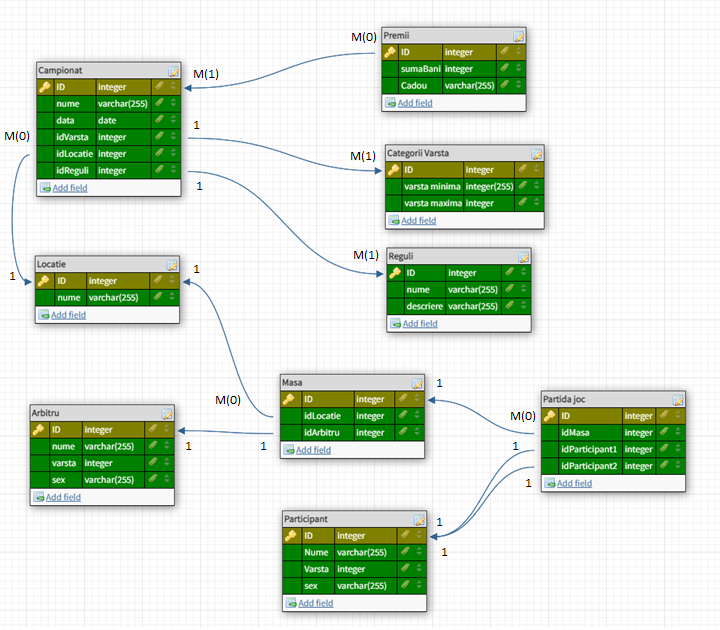
Fiecare campionat este doar pentru o categorie de varsta, astfel incat meciurile sa fie echilibrate (juniori cu juniori, seniori cu seniori, etc.) si un set de reguli necesare desfasurarii campionatului.

Partidele de joc se desfasoara intre doi participanti diferiti, la o masa din locatia x, supravegheata de arbitrul y.

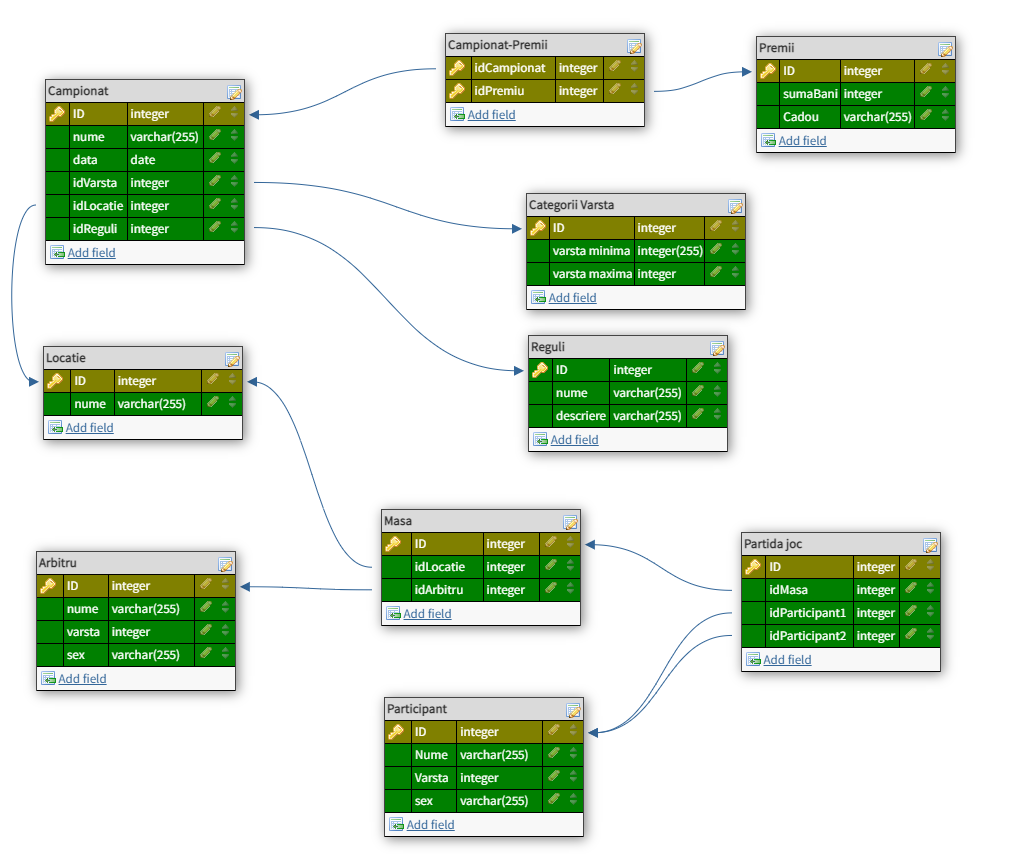
Campionatul are mai multe premii, in functie de organizatorii acestuia, premiile trebuie sa aiba un cadou (ex tricou, racheta, set de mingi), dar nu neaparat o suma de bani.

Baza de date este utila pentru a putea face conexiuni intre participanti, categoriile lor de varsta, arbitrii si premiile puse la dispozitie. Exista mai multe turnee in fiecare luna, insa nu in aceeasi zi, pentru ca fiecare participant sa poata juca la toate turneele sau sa poata fi spectator la ele, in cazul in care nu poate sa joace sau nu apartine categoriei turneului.

1. *Realizarea diagramei entitate-relatie (ERD).*



1. *Realizarea diagramei conceptuale*



**--Participant**

**CREATE TABLE TabelaParticipant(**

**idParticipant integer CONSTRAINT participant\_pk PRIMARY KEY,**

**Nume varchar(255) NOT NULL,**

**Varsta integer NOT NULL,**

**sex varchar(10) NOT NULL**

**);**

**INSERT INTO TabelaParticipant VALUES ('1','Alexe Mihai', '61', 'M');**

**INSERT INTO TabelaParticipant VALUES ('2','Bartos Diana', '44', 'M');**

**INSERT INTO TabelaParticipant VALUES ('3','Pandele Victor', '36', 'M');**

**INSERT INTO TabelaParticipant VALUES ('4','Tian Mihnea', '51', 'M');**

**INSERT INTO TabelaParticipant VALUES ('5','Dinu Matei', '31', 'M');**

**INSERT INTO TabelaParticipant VALUES ('6','Finu Prapu', '41', 'M');**

**INSERT INTO TabelaParticipant VALUES ('7','Mircea Bravo', '27', 'M');**

**INSERT INTO TabelaParticipant VALUES ('8','Melisa Andreea', '26', 'F');**

**INSERT INTO TabelaParticipant VALUES ('9','Mitu Ionut', '42', 'M');**

**INSERT INTO TabelaParticipant VALUES ('10','Dragan Mihai', '36', 'M');**

**--Arbitru**

**CREATE TABLE TabelaArbitru(**

**idArbitru integer CONSTRAINT arbitru\_pk PRIMARY KEY,**

**Nume varchar(255) NOT NULL,**

**Varsta integer NOT NULL,**

**sex varchar(10) NOT NULL**

**);**

**INSERT INTO TabelaArbitru VALUES ('1','Torin Amda', '41', 'F');**

**INSERT INTO TabelaArbitru VALUES ('2','Dan Mihai', '52', 'M');**

**INSERT INTO TabelaArbitru VALUES ('3','Pavel Florin', '37', 'M');**

**INSERT INTO TabelaArbitru VALUES ('4','Ciobanu Daniel', '28', 'M');**

**INSERT INTO TabelaArbitru VALUES ('5','Dumitru Amalia', '26', 'F');**

**INSERT INTO TabelaArbitru VALUES ('6','Ciobotaru Manuel', '44', 'M');**

**INSERT INTO TabelaArbitru VALUES ('7','Danila Prepeleac', '59', 'M');**

**INSERT INTO TabelaArbitru VALUES ('8','Delia Matache', '31', 'F');**

**INSERT INTO TabelaArbitru VALUES ('9','Cristian Siucu', '58', 'M');**

**INSERT INTO TabelaArbitru VALUES ('10','Dante Andreea', '47', 'F');**

**--Categorii de Varsta**

**CREATE TABLE TabelaCategoriiVarsta(**

**idCategorie integer CONSTRAINT categorii\_pk PRIMARY KEY,**

**varstaMinima integer NOT NULL,**

**varstaMaxima integer NOT NULL**

**);**

**INSERT INTO TabelaCategoriiVarsta VALUES ('1', '7', '14');**

**INSERT INTO TabelaCategoriiVarsta VALUES ('2', '15', '25');**

**INSERT INTO TabelaCategoriiVarsta VALUES ('3', '26', '39');**

**INSERT INTO TabelaCategoriiVarsta VALUES ('4', '40', '55');**

**INSERT INTO TabelaCategoriiVarsta VALUES ('5', '55', '70');**

**-- Mese**

**CREATE TABLE TabelaMasa(**

**idMasa integer CONSTRAINT masa\_pk PRIMARY KEY,**

**idLocatie integer NOT NULL, CONSTRAINT masa\_locatie\_fk FOREIGN KEY(idLocatie) references TabelaLocatie(idLocatie),**

**idArbitru integer NOT NULL, CONSTRAINT masa\_arbitru\_fk FOREIGN KEY(idArbitru) references TabelaArbitru(idArbitru)**

**);**

**INSERT INTO TabelaMasa VALUES (1, 1, 1);**

**INSERT INTO TabelaMasa VALUES (2, 2, 2);**

**INSERT INTO TabelaMasa VALUES (3, 3, 3);**

**INSERT INTO TabelaMasa VALUES (4, 4 ,4);**

**INSERT INTO TabelaMasa VALUES (5, 5, 5);**

**-- Reguli**

**CREATE TABLE TabelaReguli(**

**idRegula integer CONSTRAINT reguli\_pk PRIMARY KEY,**

**Nume varchar(255) NOT NULL,**

**Descriere varchar(255) NOT NULL**

**);**

**INSERT INTO TabelaReguli VALUES ('1','Regula1', 'Nu aveti voie sa jucati cu alte mingi in afara de cele oferite de organizatori');**

**INSERT INTO TabelaReguli VALUES ('2','Regula2', 'Daca pentru a ajunge la minge ati pus o mana pe masa, punctul este al adversarului');**

**INSERT INTO TabelaReguli VALUES ('3','Regula3', 'La servire, mingea trebuie aruncata vertical minim 15 centimetri');**

**INSERT INTO TabelaReguli VALUES ('4','Regula4', 'Nu este permis contactul fizic cu adversarul sau arbitrul, in afara de salutul de la inceput');**

**INSERT INTO TabelaReguli VALUES ('5','Regula5', 'Nu aveti voie serviti din lateralul mesei sau mai aproape decat marginea mesei');**

**-- Locatii**

**CREATE TABLE TabelaLocatie(**

**idLocatie integer CONSTRAINT locatie\_pk PRIMARY KEY,**

**Nume varchar(255) NOT NULL**

**);**

**INSERT INTO TabelaLocatie VALUES ('1','Bd. Ion Mihalache 255');**

**INSERT INTO TabelaLocatie VALUES ('2','Bd. Regina Elisbeta 24');**

**INSERT INTO TabelaLocatie VALUES ('3','Str. Brasov 143');**

**INSERT INTO TabelaLocatie VALUES ('4','Str. Ion Agarbiceanu 14');**

**INSERT INTO TabelaLocatie VALUES ('5','Bd. Dimitrie Gusti 317');**

**-- Premii**

**CREATE TABLE TabelaPremii(**

**idPremiu integer CONSTRAINT premii\_pk PRIMARY KEY,**

**Nume varchar(255) NOT NULL,**

**sumaBani integer NOT NULL,**

**Cadou varchar(255) NOT NULL**

**);**

**INSERT INTO TabelaPremii VALUES ('1','Premiul 1', '1000', 'Racheta Butterfly + 2 Fete Tibhar');**

**INSERT INTO TabelaPremii VALUES ('2','Premiul 2', '500', 'Racheta Nittaku');**

**INSERT INTO TabelaPremii VALUES ('3','Premiul 3', '300', '2 Fete Nittaku');**

**INSERT INTO TabelaPremii VALUES ('4','Premiul cea mai buna lovitura', '100', 'Tricou "Best shot"');**

**INSERT INTO TabelaPremii VALUES ('5','Premiul cea mai rapida minge', '0', 'Racheta carbon Xiom');**

**-- Campionate**

**CREATE TABLE TabelaCampionat(**

**idCampionat integer CONSTRAINT campionat\_pk PRIMARY KEY,**

**Nume varchar(255) NOT NULL,**

**dataCampionat date DEFAULT sysdate NOT NULL,**

**idVarsta integer NOT NULL, CONSTRAINT campionat\_categorii\_fk FOREIGN KEY(idVarsta) references TabelaCategoriiVarsta(idCategorie),**

**idLocatie integer NOT NULL, CONSTRAINT campionat\_locatie\_fk FOREIGN KEY(idLocatie) references TabelaLocatie(idLocatie),**

**idReguli integer NOT NULL, CONSTRAINT campionat\_reguli\_fk FOREIGN KEY(idReguli) references TabelaReguli(idRegula)**

**);**

**INSERT INTO TabelaCampionat VALUES (1 ,'Campionatul 1', to\_date('25-01-2023', 'DD-MM-YYYY'), 1, 1, 1);**

**INSERT INTO TabelaCampionat VALUES (2,'Campionatul 2', to\_date('28-01-2023', 'DD-MM-YYYY'), 2, 3, 5);**

**INSERT INTO TabelaCampionat VALUES (3,'Campionatul 3', to\_date('4-02-2023', 'DD-MM-YYYY'), 3, 5, 3);**

**INSERT INTO TabelaCampionat VALUES (4,'Campionatul 4', to\_date('5-02-2023', 'DD-MM-YYYY'), 5, 4, 2);**

**INSERT INTO TabelaCampionat VALUES (5,'Campionatul 5', to\_date('8-02-2023', 'DD-MM-YYYY'), 4, 2, 4);**

**-- Campionat-Premiu**

**CREATE TABLE Campionat\_Premii(**

**idCampionat integer NOT NULL, CONSTRAINT cp\_campionat\_fk FOREIGN KEY (idCampionat) references TabelaCampionat(idCampionat),**

**idPremiu integer NOT NULL, CONSTRAINT cp\_premiu\_fk FOREIGN KEY (idPremiu) references TabelaPremii(idPremiu),**

**CONSTRAINT stream\_pk PRIMARY KEY (idCampionat, idPremiu)**

**);**

**INSERT INTO Campionat\_Premii VALUES (1, 3);**

**INSERT INTO Campionat\_Premii VALUES (2, 1);**

**INSERT INTO Campionat\_Premii VALUES (3, 3);**

**INSERT INTO Campionat\_Premii VALUES (4, 2);**

**INSERT INTO Campionat\_Premii VALUES (5, 4);**

**INSERT INTO Campionat\_Premii VALUES (1, 5);**

**INSERT INTO Campionat\_Premii VALUES (2, 2);**

**INSERT INTO Campionat\_Premii VALUES (3, 1);**

**INSERT INTO Campionat\_Premii VALUES (4, 4);**

**INSERT INTO Campionat\_Premii VALUES (5, 2);**

**-- Partide de joc**

**CREATE TABLE TabelaPartidaJoc(**

**idPartidaJoc integer CONSTRAINT partida\_pk PRIMARY KEY,**

**idMasa integer NOT NULL, CONSTRAINT partida\_masa\_fk FOREIGN KEY(idMasa) references TabelaMasa(idMasa),**

**idParticipant1 integer NOT NULL, CONSTRAINT partida\_jucator1\_fk FOREIGN KEY(idParticipant1) references TabelaParticipant(idParticipant),**

**idParticipant2 integer NOT NULL, CONSTRAINT partida\_jucator2\_fk FOREIGN KEY(idParticipant2) references TabelaParticipant(idParticipant)**

**);**

**INSERT INTO TabelaPartidaJoc VALUES (1, 1, 1, 2);**

**INSERT INTO TabelaPartidaJoc VALUES (2, 2, 4, 5);**

**INSERT INTO TabelaPartidaJoc VALUES (3, 4, 3, 8);**

**INSERT INTO TabelaPartidaJoc VALUES (4, 3, 6, 7);**

**INSERT INTO TabelaPartidaJoc VALUES (5, 2, 9, 10);**

**INSERT INTO TabelaPartidaJoc VALUES (6, 5, 1, 4);**

**INSERT INTO TabelaPartidaJoc VALUES (7, 1, 3, 7);**

**INSERT INTO TabelaPartidaJoc VALUES (8, 5, 9, 1);**

**INSERT INTO TabelaPartidaJoc VALUES (9, 3, 4, 7);**

**INSERT INTO TabelaPartidaJoc VALUES (10, 2, 9, 4);**

**--6**

**-- numarul de mese la care a jucat fiecare jucator**

**CREATE OR REPLACE PROCEDURE cerinta6 IS**

**TYPE jucator\_rec IS RECORD (cod tabelaparticipant.idparticipant%TYPE, nume tabelaparticipant.nume%TYPE, nrmese tabelaparticipant.varsta%TYPE);**

**TYPE tablou IS TABLE OF jucator\_rec;**

**TYPE vector IS VARRAY(20) OF NUMBER;**

**CURSOR crs IS**

**SELECT idparticipant**

**FROM tabelaparticipant;**

**ta tablou := tablou();**

**vec vector := vector();**

**juc jucator\_rec;**

**BEGIN**

**FOR c IN crs LOOP**

**SELECT idparticipant, nume**

**INTO juc.cod, juc.nume**

**FROM tabelaparticipant**

**WHERE c.idparticipant = idparticipant;**

**vec.EXTEND;**

**SELECT COUNT (\*)**

**INTO vec(vec.LAST)**

**FROM tabelapartidajoc**

**WHERE idparticipant1 = c.idparticipant OR idparticipant2 = c.idparticipant;**

**juc.nrmese := vec(vec.LAST);**

**ta.EXTEND;**

**ta(ta.LAST) := juc;**

**IF (ta(ta.LAST).nrmese = 0) THEN**

**DBMS\_OUTPUT.PUT\_LINE ('Acest jucator nu a jucat la nicio masa');**

**ELSE**

**DBMS\_OUTPUT.PUT\_LINE ('Jucatorul ' || ta(ta.LAST).nume || ' a jucat la ' || ta(ta.LAST).nrmese || ' mese.');**

**END IF;**

**END LOOP;**

**END cerinta6;**

**/**

**EXECUTE cerinta6;**

**--7**

**-- pentru o locatie data ca parametru, mariti premiile acordate la campionatele din acea locatie cu 20%**

**CREATE OR REPLACE PROCEDURE cerinta7(atr\_locatie\_nume tabelalocatie.nume%TYPE) IS**

**CURSOR premii IS**

**SELECT tc.idlocatie, tp.idpremiu, tp.sumabani**

**FROM tabelapremii tp**

**JOIN campionat\_premii cp ON cp.idpremiu = tp.idpremiu**

**JOIN tabelacampionat tc ON tc.idcampionat = cp.idcampionat;**

**CURSOR locatii (locatie\_nume tabelalocatie.nume%TYPE) IS**

**SELECT idlocatie**

**FROM tabelalocatie tl**

**WHERE UPPER(tl.nume) LIKE UPPER(locatie\_nume);**

**v\_idpremiu tabelapremii.idpremiu%TYPE;**

**v\_bani tabelapremii.sumabani%TYPE;**

**v\_idlocatie tabelacampionat.idlocatie%TYPE;**

**BEGIN**

**FOR locatie in locatii(atr\_locatie\_nume) LOOP**

**OPEN premii;**

**LOOP**

**FETCH premii INTO v\_idlocatie, v\_idpremiu, v\_bani;**

**EXIT WHEN premii%NOTFOUND;**

**IF v\_idlocatie = locatie.idlocatie**

**THEN**

**UPDATE tabelapremii tp**

**SET tp.sumabani = tp.sumabani + ((tp.sumabani \* 20)/100)**

**WHERE v\_idpremiu = tp.idpremiu;**

**END IF;**

**END LOOP;**

**END LOOP;**

**END;**

**/**

**SELECT \* FROM tabelapremii;**

**CALL cerinta7('Str. Brasov 143');**

**--8**

**--Sa se afiseze numarul de meciuri pe care le-a jucat participantul x cu arbitrul y. Daca nu exista participantul sau nu a jucat niciun meci, eroare.**

**CREATE OR REPLACE FUNCTION cerinta8 (input\_name tabelaparticipant.nume%TYPE, id\_arbitru tabelamasa.idarbitru%TYPE)**

**RETURN NUMBER IS**

**counter NUMBER;**

**no\_participants EXCEPTION;**

**too\_many\_participants EXCEPTION;**

**no\_games\_played EXCEPTION;**

**BEGIN**

**SELECT COUNT(\*) INTO counter**

**FROM tabelaparticipant**

**WHERE LOWER(input\_name) = LOWER (nume);**

**IF (counter = 0)**

**THEN RAISE no\_participants;**

**ELSIF (counter > 1)**

**THEN RAISE too\_many\_participants;**

**END IF;**

**SELECT COUNT(\*) INTO counter**

**FROM tabelapartidajoc tpc**

**INNER JOIN tabelamasa tm ON tpc.idmasa = tm.idmasa**

**INNER JOIN tabelaparticipant tp ON (tp.idparticipant = tpc.idparticipant1 OR tp.idparticipant = tpc.idparticipant2)**

**WHERE LOWER(tp.nume) = LOWER(input\_name) AND tm.idarbitru = id\_arbitru;**

**IF (counter = 0)**

**THEN RAISE no\_games\_played;**

**END IF;**

**RETURN counter;**

**EXCEPTION**

**WHEN no\_participants THEN**

**DBMS\_OUTPUT.PUT\_LINE ('Nu exista participantul dat ca parametru.');**

**RETURN -1;**

**WHEN too\_many\_participants THEN**

**DBMS\_OUTPUT.PUT\_LINE('Exista cel putin 2 participanti cu acelasi nume.');**

**RETURN -2;**

**WHEN no\_games\_played THEN**

**DBMS\_OUTPUT.PUT\_LINE('Acest jucator nu a jucat niciun meci.');**

**RETURN -3;**

**END cerinta8;**

**/**

**SELECT cerinta8('Alexe Mihai', 5) FROM DUAL;**

**--9**

**--Sa se afiseze numarul de meciuri pe care le-a jucat participantul x cu arbitrul y, la locatia z . Daca nu exista participantul sau nu a jucat niciun meci, eroare.**

**CREATE OR REPLACE PROCEDURE cerinta9 (input\_name tabelaparticipant.nume%TYPE, nume\_arbitru tabelaarbitru.nume%TYPE, nume\_locatie tabelalocatie.nume%TYPE) IS**

**numePart tabelaparticipant.nume%TYPE;**

**counter INTEGER;**

**BEGIN**

**SELECT nume INTO numePart**

**FROM tabelaparticipant**

**WHERE LOWER(input\_name) = LOWER (nume);**

**-- IF (counter = 0)**

**-- THEN RAISE\_APPLICATION\_ERROR(-20022, 'Nu exista participantul dat ca parametru.');**

**-- ELSIF (counter > 1)**

**-- THEN RAISE\_APPLICATION\_ERROR ( -20022 , 'Exista cel putin 2 participanti cu acelasi nume.');**

**-- END IF;**

**--**

**SELECT COUNT(\*) INTO counter**

**FROM tabelapartidajoc tpc**

**JOIN tabelamasa tm ON tpc.idmasa = tm.idmasa**

**JOIN tabelaparticipant tp ON (tp.idparticipant = tpc.idparticipant1 OR tp.idparticipant = tpc.idparticipant2)**

**JOIN tabelaarbitru ta ON ta.idarbitru = tm.idarbitru**

**JOIN tabelalocatie tl ON tl.idlocatie = tm.idlocatie**

**WHERE LOWER(tp.nume) = LOWER(input\_name) AND**

**ta.nume = nume\_arbitru AND**

**tl.nume = nume\_locatie;**

**DBMS\_OUTPUT.PUT\_LINE ('Participantul cautat a jucat la ' || counter || ' mese');**

**EXCEPTION**

**WHEN TOO\_MANY\_ROWS THEN**

**RAISE\_APPLICATION\_ERROR ( -20022 , 'Exista cel putin 2 participanti cu acelasi nume.');**

**WHEN NO\_DATA\_FOUND THEN**

**RAISE\_APPLICATION\_ERROR(-20022, 'Nu exista participantul dat ca parametru.');**

**-- IF (counter = 0)**

**-- THEN RAISE\_APPLICATION\_ERROR (-20024, 'Acest jucator nu a jucat niciun meci cu arbitrul ' || nume\_arbitru);**

**-- END IF;**

**END cerinta9;**

**/**

**EXECUTE cerinta9('Alexe Mihai', 'Dumitru Amalia', 'Bd. Dimitrie Gusti 317');**

**--10.**

**--Trigger LMD la nivel de comanda**

**-- Trigger-ul se declanseaza daca un campionat nu este finalizat si se incearca decernarea premiilor.**

**CREATE OR REPLACE TRIGGER cerinta10**

**BEFORE INSERT OR UPDATE OR DELETE ON Campionat\_Premii**

**BEGIN**

**IF (SYSDATE BETWEEN to\_date('25-01-2023', 'DD-MM-YYYY') AND to\_date('8-02-2023', 'DD-MM-YYYY')) THEN**

**RAISE\_APPLICATION\_ERROR(-20041, 'Un campionat este in desfasurare, nu se pot oferi premii pana la finalizarea tuturor meciurilor.');**

**END IF;**

**END;**

**INSERT INTO Campionat\_Premii VALUES (2,4);**

**-- 11**

**-- Trigger de tip LMD la nivel de linie**

**-- Se declanseaza cand se incearca introducerea / actualizarea unui meci in care echipa gazda este aceeasi cu cea oaspete**

**CREATE OR REPLACE TRIGGER echipe\_diferite**

**BEFORE INSERT OR UPDATE ON TabelaPartidaJoc**

**FOR EACH ROW**

**BEGIN**

**IF (:NEW.idParticipant1 = :NEW.idParticipant2)**

**THEN RAISE\_APPLICATION\_ERROR(-20022, 'Meciul trebuie sa se desfasoare intre jucatori diferiti');**

**END IF;**

**END;**

**CREATE SEQUENCE SEQ\_PARTIDE start with 10 increment by 1;**

**INSERT INTO TabelaPartidaJoc VALUES (SEQ\_PARTIDE.nextval, 2, 3, 3);**

**CREATE OR REPLACE TRIGGER varsta\_minima**

**BEFORE INSERT OR UPDATE ON TabelaArbitru**

**FOR EACH ROW**

**BEGIN**

**IF (:NEW.varsta < 25)**

**THEN RAISE\_APPLICATION\_ERROR(-20023, 'Un arbitru nu poate avea mai putin de 25 ani');**

**END IF;**

**END;**

**CREATE SEQUENCE SEQ\_ARBITRI start with 10 increment by 1;**

**INSERT INTO TabelaArbitru VALUES (SEQ\_ARBITRI.nextval, 'Bernard Mattias', 22, 'M');**

**-- Cerinta 12**

**-- Trigger care afiseaza un mesaj in momentul rularii unei comenzi de tip LDD**

**CREATE OR REPLACE TRIGGER cerinta12\_LDD**

**AFTER CREATE OR ALTER OR DROP ON SCHEMA**

**BEGIN**

**DBMS\_OUTPUT.PUT\_LINE('S-a rulat o comanda de tip LDD');**

**END;**

**/**

**SET SERVEROUTPUT ON;**

**ALTER TABLE TabelaParticipant**

**DROP COLUMN CampionateCastigate;**

**ALTER TABLE TabelaParticipant**

**ADD CampionateCastigate INTEGER;**