**Hutan Mihai-Alexandru**

**Grupa 243**

**Gestiunea competitiei Esports de League of Legends**

|  |
| --- |
| EX1 |

**Prezentati pe scurt baza de date (utilitatea ei).**

**Baza de date contine informatii cu privire la turneele acestei competitii, grupele si echipele participante, cat si jucatorii, ce meciuri au acestia, prezentatorii, scena si modul de joc al meciurilor. Apar detalii legate de jucatori si echipa. Jucatorii si echipele apartin unei regiuni, iar jucatorii joaca in mod frecvent anumiti campioni salvati in baza de date.**

**Scopul crearii acestei baze de date este de a facilita generarea de statistici, ce ulterior vor fi analizate in cadrul evenimentelor de tip Esports.**

**Aceasta competitie are un numar foarte mare de turnee an de an. Fiecare turneu continte informatii relevante precum numele, orasul si data inceperii acestuia si este identificat printr-un id unic.**

**Un turneu contine mai multe grupe identificate printr-un id unic, ce au si o denumire.**

**Fiecare grupa contine mai multe echipe despre care cunoastem numele, tag-ul, id-ul si regiunea din care provin.**

**Regiunea reprezinta un atribut si in cazul jucatorilor despre care mai cunoastem numele, lane-ul si id-ul, cat si campionii pe care ii joaca in mod frecvent, care la randul lor au un id unic, un titlu si resursa folosita.**

**Fiind un turneu de competitie Esports, vor avea loc meciuri, care vor fi comentate de diferiti prezentatori, vor avea loc intre diferite echipe, moduri de joc multiple. Cunoscand toate aceste detalii putem crea un MeciHeader ce va ajuta echipa de marketing pentru a promova meciurile importante.**

|  |
| --- |
| EX2 |

**Realizati diagrama entitate-relatie (ERD).**

**Diagram

Description automatically generated**

|  |
| --- |
| EX3 |

**Pornind de la diagrama entitate-relatie realizați diagrama conceptuala a modelului propus, integrand toate atributele necesare.**

**Diagram

Description automatically generated**

**----------------------------------------------------------------------**

**Diagram

Description automatically generated**

|  |
| --- |
| EX4 & EX5 |

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

***----------------------------------- SCENA -----------------------------------***

**CREATE TABLE scena**

**(**

**id NUMBER(5) GENERATED ALWAYS AS IDENTITY**

***CONSTRAINT* pkey\_scena PRIMARY KEY,**

**nume VARCHAR2(50)**

***CONSTRAINT* nume\_scena NOT NULL**

**)**

**INSERT INTO scena(nume)**

**VALUES ('Main Stage')**

**INSERT INTO scena(nume)**

**VALUES ('Second Stage')**

**INSERT INTO scena(nume)**

**VALUES ('Third Stage')**

**INSERT INTO scena(nume)**

**VALUES ('Horn Stage')**

**INSERT INTO scena(nume)**

**VALUES ('Panther Stage')**

**COMMIT;**

***----------------------------------- PREZENTATOR -----------------------------------***

**CREATE TABLE prezentator**

**(**

**id NUMBER(5) GENERATED ALWAYS AS IDENTITY**

***CONSTRAINT* pkey\_prezentator PRIMARY KEY,**

**nume VARCHAR2(50)**

***CONSTRAINT* nume\_prezentator NOT NULL**

**)**

**INSERT INTO prezentator(nume)**

**VALUES ('James Patterson')**

**INSERT INTO prezentator(nume)**

**VALUES ('David Turley')**

**INSERT INTO prezentator(nume)**

**VALUES ('Sam Hartman')**

**INSERT INTO prezentator(nume)**

**VALUES ('Mark Zimmerman')**

**INSERT INTO prezentator(nume)**

**VALUES ('Emily Rand')**

**INSERT INTO prezentator(nume)**

**VALUES ('Barento Mohammed')**

**INSERT INTO prezentator(nume)**

**VALUES ('Julian Carr')**

**INSERT INTO prezentator(nume)**

**VALUES ('Isaac Bentley')**

**INSERT INTO prezentator(nume)**

**VALUES ('Max Anderson')**

**INSERT INTO prezentator(nume)**

**VALUES ('Jordan Corby')**

**COMMIT;**

***----------------------------------- MOD\_JOC -----------------------------------***

**CREATE TABLE mod\_joc**

**(**

**id NUMBER(5) GENERATED ALWAYS AS IDENTITY**

***CONSTRAINT* pkey\_mod\_joc PRIMARY KEY,**

**nume VARCHAR2(50)**

***CONSTRAINT* nume\_mod\_joc NOT NULL,**

**harta VARCHAR2(50)**

***CONSTRAINT* harta\_mod\_joc NOT NULL,**

**dimensiune NUMBER(5)**

***CONSTRAINT* dimensiune\_mod\_joc NOT NULL**

**)**

**INSERT INTO mod\_joc(nume, harta, dimensiune)**

**VALUES ('Normal', 'Summoners Rift', '5')**

**INSERT INTO mod\_joc(nume, harta, dimensiune)**

**VALUES ('Ranked', 'Summoners Rift', '5')**

**INSERT INTO mod\_joc(nume, harta, dimensiune)**

**VALUES ('Flex', 'Summoners Rift', '5')**

**INSERT INTO mod\_joc(nume, harta, dimensiune)**

**VALUES ('Normal', 'Summoners Rift', '5')**

**INSERT INTO mod\_joc(nume, harta, dimensiune)**

**VALUES ('Normal', 'Twisted Treelines', '3')**

**INSERT INTO mod\_joc(nume, harta, dimensiune)**

**VALUES ('Ranked', 'Twisted Treelines', '3')**

**INSERT INTO mod\_joc(nume, harta, dimensiune)**

**VALUES ('Flex', 'Twisted Treelines', '3')**

**INSERT INTO mod\_joc(nume, harta, dimensiune)**

**VALUES ('Normal', 'ARAM', '5')**

**INSERT INTO mod\_joc(nume, harta, dimensiune)**

**VALUES ('Ranked', 'ARAM', '5')**

**INSERT INTO mod\_joc(nume, harta, dimensiune)**

**VALUES ('Flex', 'ARAM', '5')**

**INSERT INTO mod\_joc(nume, harta, dimensiune)**

**VALUES ('Ranked', 'ARAM', '1')**

**COMMIT;**

***----------------------------------- MECI -----------------------------------***

**CREATE TABLE meci**

**(**

**id NUMBER(5) GENERATED ALWAYS AS IDENTITY**

***CONSTRAINT* pkey\_meci PRIMARY KEY,**

**titlu VARCHAR2(50)**

***CONSTRAINT* titlu\_meci NOT NULL,**

**data DATE**

***CONSTRAINT* data\_meci NOT NULL**

**)**

**INSERT INTO meci(titlu, data)**

**VALUES ('Revenge', TO\_DATE('2022-05-10', 'YYYY-MM-DD'))**

**INSERT INTO meci(titlu, data)**

**VALUES ('Revenge', TO\_DATE('2022-05-11', 'YYYY-MM-DD'))**

**INSERT INTO meci(titlu, data)**

**VALUES ('Warriors', TO\_DATE('2022-05-13', 'YYYY-MM-DD'))**

**INSERT INTO meci(titlu, data)**

**VALUES ('Destroyers', TO\_DATE('2022-05-14', 'YYYY-MM-DD'))**

**INSERT INTO meci(titlu, data)**

**VALUES ('Debut', TO\_DATE('2021-10-01', 'YYYY-MM-DD'))**

**INSERT INTO meci(titlu, data)**

**VALUES ('Academy', TO\_DATE('2021-10-02', 'YYYY-MM-DD'))**

**INSERT INTO meci(titlu, data)**

**VALUES ('Academy', TO\_DATE('2021-05-03', 'YYYY-MM-DD'))**

**INSERT INTO meci(titlu, data)**

**VALUES ('RunnerUps', TO\_DATE('2022-01-12', 'YYYY-MM-DD'))**

**INSERT INTO meci(titlu, data)**

**VALUES ('RunnerUps', TO\_DATE('2022-01-12', 'YYYY-MM-DD'))**

**INSERT INTO meci(titlu, data)**

**VALUES ('Academy', TO\_DATE('2021-01-13', 'YYYY-MM-DD'))**

**COMMIT;**

***----------------------------------- CAMPION -----------------------------------***

**CREATE TABLE campion**

**(**

**id NUMBER(5) GENERATED ALWAYS AS IDENTITY**

***CONSTRAINT* pkey\_campion PRIMARY KEY,**

**nume VARCHAR2(50)**

***CONSTRAINT* nume\_campion NOT NULL,**

**titlu VARCHAR2(50)**

***CONSTRAINT* titlu\_campion NOT NULL,**

**resursa VARCHAR2(50)**

***CONSTRAINT* resursa\_campion NOT NULL**

**)**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Aatrox', 'the Darkin Blade', 'Blood Well')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Ahri', 'the Nine-Tailed Fox', 'Mana')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Akali', 'the Rogue Assassin', 'Energy')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Alistar', 'the Minotaur', 'Mana')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Amumu', 'the Sad Mummy', 'Mana')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Anivia', 'the Cryophoenix', 'Mana')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Annie', 'the Dark Child', 'Mana')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Ashe', 'the Frost Archer', 'Mana')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Fiddlesticks', 'the Ancient Fear', 'Mana')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Fizz', 'the Tidal Trickster', 'Mana')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Gangplank', 'the Saltwater Scourge', 'Mana')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Hecarim', 'the Shadow of War', 'Mana')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Jax', 'Grandmaster at Arms', 'Mana')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Jayce', 'the Defender of Tomorrow', 'Mana')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Jhin', 'the Virtuoso', 'Mana')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Jinx', 'the Loose Cannon', 'Mana')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('KhaZix', 'the Voidreaver', 'Mana')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Lee Sin', 'the Blind Monk', 'Energy')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Zed', 'the Master of Shadows', 'Energy')**

**INSERT INTO campion(nume, titlu, resursa)**

**VALUES ('Lux', 'the Lady of Luminosity', 'Mana')**

**COMMIT;**

***----------------------------------- TURNEU -----------------------------------***

**CREATE TABLE turneu**

**(**

**id NUMBER(5) GENERATED ALWAYS AS IDENTITY**

***CONSTRAINT* pkey\_turneu PRIMARY KEY,**

**nume VARCHAR2(50)**

***CONSTRAINT* nume\_turneu NOT NULL,**

**oras VARCHAR2(50)**

***CONSTRAINT* oras\_turneu NOT NULL,**

**data DATE**

***CONSTRAINT* data\_turneu NOT NULL**

**)**

**INSERT INTO turneu(nume, oras, data)**

**VALUES ('MSI', 'Roma', TO\_DATE('2022-05-10', 'YYYY-MM-DD'))**

**INSERT INTO turneu(nume, oras, data)**

**VALUES ('ALLSTARS', 'New York', TO\_DATE('2022-01-12', 'YYYY-MM-DD'))**

**INSERT INTO turneu(nume, oras, data)**

**VALUES ('WORLDS', 'Beijing', TO\_DATE('2021-10-01', 'YYYY-MM-DD'))**

**INSERT INTO turneu(nume, oras, data)**

**VALUES ('LCK-SUMMER', 'Seul', TO\_DATE('2021-06-05', 'YYYY-MM-DD'))**

**INSERT INTO turneu(nume, oras, data)**

**VALUES ('NA-SPRING', 'Boston', TO\_DATE('2021-03-13', 'YYYY-MM-DD'))**

**INSERT INTO turneu(nume, oras, data)**

**VALUES ('EUW-SUMMER', 'Paris', TO\_DATE('2022-06-05', 'YYYY-MM-DD'))**

**INSERT INTO turneu(nume, oras, data)**

**VALUES ('NA-SUMMER', 'Los Angeles', TO\_DATE('2022-07-03', 'YYYY-MM-DD'))**

**INSERT INTO turneu(nume, oras, data)**

**VALUES ('EUW-SPRING', 'London', TO\_DATE('2022-04-01', 'YYYY-MM-DD'))**

**INSERT INTO turneu(nume, oras, data)**

**VALUES ('LCK-SPRING', 'Seul', TO\_DATE('2021-03-05', 'YYYY-MM-DD'))**

**INSERT INTO turneu(nume, oras, data)**

**VALUES ('OCE-SUMMER', 'Sydney', TO\_DATE('2022-07-01', 'YYYY-MM-DD'))**

**COMMIT;**

***----------------------------------- REGIUNE -----------------------------------***

**CREATE TABLE regiune**

**(**

**id NUMBER(5) GENERATED ALWAYS AS IDENTITY**

***CONSTRAINT* pkey\_regiune PRIMARY KEY,**

**nume VARCHAR2(50)**

***CONSTRAINT* nume\_regiune NOT NULL**

**)**

**INSERT INTO regiune(nume)**

**VALUES ('KOR')**

**INSERT INTO regiune(nume)**

**VALUES ('EUNE')**

**INSERT INTO regiune(nume)**

**VALUES ('EUW')**

**INSERT INTO regiune(nume)**

**VALUES ('NA')**

**INSERT INTO regiune(nume)**

**VALUES ('OCE')**

**INSERT INTO regiune(nume)**

**VALUES ('SA')**

**INSERT INTO regiune(nume)**

**VALUES ('TUR')**

**INSERT INTO regiune(nume)**

**VALUES ('RUS')**

**COMMIT;**

***----------------------------------- GRUPA -----------------------------------***

**CREATE TABLE grupa**

**(**

**id NUMBER(5) GENERATED ALWAYS AS IDENTITY**

***CONSTRAINT* pkey\_grupa PRIMARY KEY,**

**nume VARCHAR2(50)**

***CONSTRAINT* nume\_grupa NOT NULL,**

**id\_turneu NUMBER(5) NOT NULL,**

***CONSTRAINT* fk\_turneu\_grupa *FOREIGN KEY* (id\_turneu) *REFERENCES* turneu (id)**

**)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('A', 1)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('B', 1)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('C', 1)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('D', 1)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('A', 2)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('B', 2)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('C', 2)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('D', 2)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('A', 3)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('B', 3)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('C', 3)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('D', 3)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('A', 4)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('B', 4)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('C', 4)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('D', 4)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('A', 5)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('B', 5)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('C', 5)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('D', 5)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('A', 6)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('B', 6)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('C', 6)**

**INSERT INTO grupa(nume, id\_turneu)**

**VALUES ('D', 6)**

**COMMIT;**

***----------------------------------- ECHIPA -----------------------------------***

**CREATE TABLE echipa**

**(**

**id NUMBER(5) GENERATED ALWAYS AS IDENTITY**

***CONSTRAINT* pkey\_echipa PRIMARY KEY,**

**nume VARCHAR2(50)**

***CONSTRAINT* nume\_echipa NOT NULL,**

**tag VARCHAR2(50)**

***CONSTRAINT* tag\_echipa NOT NULL,**

**id\_regiune NUMBER(5) NOT NULL,**

***CONSTRAINT* fk\_regiune\_echipa *FOREIGN KEY* (id\_regiune) *REFERENCES* regiune (id)**

**)**

**INSERT INTO echipa(id\_regiune, nume, tag)**

**VALUES (1, 'SKT-T1', 'SKT')**

**INSERT INTO echipa(id\_regiune, nume, tag)**

**VALUES (1, 'Royal Never Give Up', 'RNG')**

**INSERT INTO echipa(id\_regiune, nume, tag)**

**VALUES (3, 'G2 Esports', 'G2')**

**INSERT INTO echipa(id\_regiune, nume, tag)**

**VALUES (4, 'Evil Geniuses ', 'EG')**

**INSERT INTO echipa(id\_regiune, nume, tag)**

**VALUES (6, 'PSG Talon Esports', 'PSG')**

**INSERT INTO echipa(id\_regiune, nume, tag)**

**VALUES (6, 'Canids Kalunga', 'RED')**

**INSERT INTO echipa(id\_regiune, nume, tag)**

**VALUES (1, 'Team Aze', 'AZE')**

**INSERT INTO echipa(id\_regiune, nume, tag)**

**VALUES (5, 'Order', 'ORD')**

**INSERT INTO echipa(id\_regiune, nume, tag)**

**VALUES (7, 'fastpayWildcats', 'IW')**

**INSERT INTO echipa(id\_regiune, nume, tag)**

**VALUES (8, 'SaigonBuffaloEsports', 'SGB')**

**COMMIT;**

***----------------------------------- GRUPA\_MAP -----------------------------------***

**CREATE TABLE grupa\_map**

**(**

**id\_grupa NUMBER(5)**

***CONSTRAINT* pk\_grupa\_map\_grupa *REFERENCES* grupa (id),**

**id\_echipa NUMBER(5)**

***CONSTRAINT* pk\_grupa\_map\_echipa *REFERENCES* echipa (id),**

***CONSTRAINT* pk\_grupa\_map PRIMARY KEY (id\_grupa, id\_echipa)**

**)**

**INSERT INTO grupa\_map(id\_grupa, id\_echipa)**

**VALUES (1, 1)**

**INSERT INTO grupa\_map(id\_grupa, id\_echipa)**

**VALUES (1, 2)**

**INSERT INTO grupa\_map(id\_grupa, id\_echipa)**

**VALUES (1, 3)**

**INSERT INTO grupa\_map(id\_grupa, id\_echipa)**

**VALUES (1, 4)**

**INSERT INTO grupa\_map(id\_grupa, id\_echipa)**

**VALUES (2, 5)**

**INSERT INTO grupa\_map(id\_grupa, id\_echipa)**

**VALUES (2, 6)**

**INSERT INTO grupa\_map(id\_grupa, id\_echipa)**

**VALUES (2, 7)**

**INSERT INTO grupa\_map(id\_grupa, id\_echipa)**

**VALUES (2, 8)**

**INSERT INTO grupa\_map(id\_grupa, id\_echipa)**

**VALUES (3, 9)**

**INSERT INTO grupa\_map(id\_grupa, id\_echipa)**

**VALUES (3, 10)**

**COMMIT;**

***----------------------------------- JUCATOR -----------------------------------***

**CREATE TABLE jucator**

**(**

**id NUMBER(5) GENERATED ALWAYS AS IDENTITY**

***CONSTRAINT* pkey\_jucator PRIMARY KEY,**

**nume VARCHAR2(50)**

***CONSTRAINT* nume\_jucator NOT NULL,**

**lane VARCHAR2(50)**

***CONSTRAINT* lane\_jucator NOT NULL,**

**id\_echipa NUMBER(5) NOT NULL,**

***CONSTRAINT* fk\_echipa\_jucator *FOREIGN KEY* (id\_echipa) *REFERENCES* echipa (id),**

**id\_regiune NUMBER(5) NOT NULL,**

***CONSTRAINT* fk\_regiune\_jucator *FOREIGN KEY* (id\_regiune) *REFERENCES* regiune (id)**

**)**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (1, 1, 'Zeus', 'TOP')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (1, 1, 'Oner', 'JUNGLE')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (1, 1, 'Faker', 'MID')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (1, 1, 'Gumayusi', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (1, 1, 'Keria', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (2, 1, 'Bin', 'TOP')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (2, 1, 'Wei', 'JUNGLE')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (2, 1, 'Xiaohu', 'MID')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (2, 1, 'Gala', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (2, 1, 'Ming', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (3, 3, 'BrokenBlade', 'TOP')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (3, 2, 'Jankos', 'JUNGLE')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (3, 3, 'Caps', 'MID')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (3, 3, 'Flakked', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (3, 3, 'Targamas', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (4, 1, 'Impact', 'TOP')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (4, 2, 'Inspired', 'JUNGLE')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (4, 4, 'Jojopyun', 'MID')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (4, 4, 'Danny', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (4, 4, 'Vulcan', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (5, 5, 'Hanabi', 'TOP')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (5, 1, 'Juhan', 'JUNGLE')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (5, 1, 'Bay', 'MID')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (5, 5, 'Unified', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (5, 5, 'Kaiwing', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (6, 6, 'Guigo', 'TOP')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (6, 6, 'Aegis', 'JUNGLE')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (6, 6, 'Avenger', 'MID')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (6, 6, 'TitaN', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (6, 6, 'Jojo', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (7, 1, 'Lonely', 'TOP')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (7, 6, 'Dimitry', 'JUNGLE')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (7, 6, 'Aloned', 'MID')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (7, 1, '5Kid', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (7, 6, 'Straight', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (8, 5, 'BioPanther', 'TOP')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (8, 5, 'Maximize', 'JUNGLE')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (8, 5, 'Kisee', 'MID')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (8, 5, 'Puma', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (8, 5, 'Corporal', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (9, 7, 'StarScreen', 'TOP')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (9, 7, 'Ferret', 'JUNGLE')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (9, 7, 'Serin', 'MID')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (9, 7, 'HolyPhoenix', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (9, 7, 'Farfetch', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (10, 8, 'Hasmed', 'TOP')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (10, 8, 'Bean J', 'JUNGLE')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (10, 8, 'Froggy', 'MID')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (10, 8, 'Shogun', 'BOT')**

**INSERT INTO jucator(id\_echipa, id\_regiune, nume, lane)**

**VALUES (10, 8, 'Taki', 'BOT')**

**COMMIT;**

***----------------------------------- CAMPION\_POOL -----------------------------------***

**CREATE TABLE campion\_pool**

**(**

**id\_campion NUMBER(5)**

***CONSTRAINT* pk\_campion\_pool\_campion *REFERENCES* campion (id),**

**id\_jucator NUMBER(5)**

***CONSTRAINT* pk\_campion\_pool\_jucator *REFERENCES* jucator (id),**

***CONSTRAINT* pk\_campion\_pool PRIMARY KEY (id\_campion, id\_jucator)**

**)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (1, 1)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (1, 3)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (1, 5)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (2, 2)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (2, 4)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (2, 6)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (3, 2)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (3, 3)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (3, 6)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (5, 18)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (4, 20)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (4, 11)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (5, 5)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (6, 10)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (7, 11)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (8, 13)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (8, 14)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (8, 15)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (9, 19)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (9, 20)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (10, 18)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (13, 9)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (12, 10)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (11, 11)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (12, 12)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (13, 13)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (14, 14)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (15, 15)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (16, 16)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (15, 17)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (16, 18)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (17, 19)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (17, 20)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (18, 2)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (18, 3)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (17, 18)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (18, 4)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (19, 5)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (20, 6)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (19, 7)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (21, 8)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (21, 9)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (20, 10)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (21, 13)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (22, 12)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (22, 11)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (23, 14)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (23, 15)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (23, 16)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (24, 19)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (24, 20)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (25, 18)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (26, 17)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (27, 16)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (28, 15)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (29, 14)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (30, 13)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (31, 12)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (32, 11)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (33, 10)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (34, 9)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (35, 8)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (36, 7)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (37, 6)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (38, 5)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (39, 4)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (40, 3)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (41, 2)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (42, 1)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (43, 20)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (44, 19)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (45, 18)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (46, 17)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (47, 16)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (48, 15)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (49, 14)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (50, 13)**

**INSERT INTO campion\_pool(id\_jucator, id\_campion)**

**VALUES (50, 12)**

**COMMIT;**

***----------------------------------- MECI\_MAP -----------------------------------***

**CREATE TABLE meci\_map**

**(**

**id\_echipa NUMBER(5)**

***CONSTRAINT* pk\_meci\_map\_echipa *REFERENCES* echipa (id),**

**id\_meci NUMBER(5)**

***CONSTRAINT* pk\_meci\_map\_meci *REFERENCES* meci (id),**

***CONSTRAINT* pk\_meci\_map PRIMARY KEY (id\_echipa, id\_meci)**

**)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (1, 1)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (2, 1)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (3, 2)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (4, 2)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (1, 3)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (3, 3)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (1, 4)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (4, 4)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (2, 5)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (3, 5)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (5, 6)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (6, 6)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (5, 7)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (7, 7)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (6, 8)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (7, 8)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (8, 9)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (5, 9)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (5, 10)**

**INSERT INTO meci\_map(id\_echipa, id\_meci)**

**VALUES (6, 10)**

**COMMIT;**

***----------------------------------- MECI\_HEADER -----------------------------------***

**CREATE TABLE meci\_header**

**(**

**id\_meci NUMBER(5)**

***CONSTRAINT* pk\_meci\_header\_meci *REFERENCES* meci (id),**

**id\_prezentator NUMBER(5)**

***CONSTRAINT* pk\_meci\_header\_prezentator *REFERENCES* prezentator (id),**

**id\_scena NUMBER(5)**

***CONSTRAINT* pk\_meci\_header\_scena *REFERENCES* scena (id),**

**id\_mod NUMBER(5)**

***CONSTRAINT* pk\_meci\_header\_mod *REFERENCES* mod\_joc (id),**

***CONSTRAINT* pk\_meci\_header PRIMARY KEY (id\_meci, id\_prezentator, id\_scena, id\_mod)**

**)**

**INSERT INTO meci\_header(id\_meci, id\_prezentator, id\_scena, id\_mod)**

**VALUES (1, 1, 3, 1)**

**INSERT INTO meci\_header(id\_meci, id\_prezentator, id\_scena, id\_mod)**

**VALUES (2, 2, 1, 3)**

**INSERT INTO meci\_header(id\_meci, id\_prezentator, id\_scena, id\_mod)**

**VALUES (3, 4, 2, 7)**

**INSERT INTO meci\_header(id\_meci, id\_prezentator, id\_scena, id\_mod)**

**VALUES (4, 7, 3, 7)**

**INSERT INTO meci\_header(id\_meci, id\_prezentator, id\_scena, id\_mod)**

**VALUES (5, 9, 2, 10)**

**INSERT INTO meci\_header(id\_meci, id\_prezentator, id\_scena, id\_mod)**

**VALUES (6, 10, 3, 5)**

**INSERT INTO meci\_header(id\_meci, id\_prezentator, id\_scena, id\_mod)**

**VALUES (7, 3, 1, 4)**

**INSERT INTO meci\_header(id\_meci, id\_prezentator, id\_scena, id\_mod)**

**VALUES (8, 6, 1, 8)**

**INSERT INTO meci\_header(id\_meci, id\_prezentator, id\_scena, id\_mod)**

**VALUES (9, 8, 3, 9)**

**INSERT INTO meci\_header(id\_meci, id\_prezentator, id\_scena, id\_mod)**

**VALUES (10, 10, 4, 10)**

**COMMIT**

|  |
| --- |
| EX6 |

**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.**

***-- Cerința 6:***

***-- Pentru un turneu al carui nume este dat, pentru fiecare regiune, afisati toate echipele si jucatorii ce apartin* *acelei regiuni sau mesajul 'Nu exista!'.***

**CREATE OR REPLACE PROCEDURE ex6(nume\_turneu turneu.nume%TYPE)**

**AS**

**TYPE tabl\_idx IS TABLE OF regiune%ROWTYPE INDEX BY PLS\_INTEGER;**

**v\_regiuni tabl\_idx;**

**TYPE tip\_lista\_nested IS TABLE OF echipa%ROWTYPE;**

**v\_echipe tip\_lista\_nested := tip\_lista\_nested();**

**TYPE tabl\_index IS TABLE OF VARCHAR2(200) INDEX BY PLS\_INTEGER;**

**v\_nume tabl\_index;**

**v\_numar NUMBER(5);**

**BEGIN**

**SELECT \* BULK COLLECT INTO v\_regiuni FROM regiune;**

**SELECT COUNT(\*)**

**INTO v\_numar**

**FROM echipa e,**

**turneu t,**

**grupa\_map gm,**

**grupa g**

**WHERE e.id = gm.id\_echipa**

**AND t.id = gm.id\_grupa**

**AND g.id = gm.id\_grupa**

**AND t.id = g.id\_turneu**

**AND UPPER(t.nume) LIKE UPPER(nume\_turneu);**

**v\_echipe.extend(v\_numar + 1);**

**SELECT e.id, e.nume, e.tag, e.id\_regiune BULK COLLECT**

**INTO v\_echipe**

**FROM echipa e,**

**turneu t,**

**grupa\_map gm,**

**grupa g**

**WHERE e.id = gm.id\_echipa**

**AND t.id = gm.id\_grupa**

**AND g.id = gm.id\_grupa**

**AND t.id = g.id\_turneu**

**AND UPPER(t.nume) LIKE UPPER(nume\_turneu);**

**FOR i IN v\_regiuni.first..v\_regiuni.last**

**LOOP**

**dbms\_output.put\_line('REGIUNEA: ' || v\_regiuni(i).nume);**

**dbms\_output.put\_line('---------------------------------');**

**FOR j IN v\_echipe.first..v\_echipe.last**

**LOOP**

**dbms\_output.put\_line('ECHIPA: ' || v\_echipe(j).nume);**

**dbms\_output.put\_line('---------------------------------');**

**SELECT j.nume BULK COLLECT**

**INTO v\_nume**

**FROM jucator j,**

**echipa e,**

**regiune r**

**WHERE j.id\_echipa = e.id**

**AND j.id\_regiune = r.id**

**AND e.nume = v\_echipe(j).nume**

**AND r.nume = v\_regiuni(i).nume;**

**IF v\_nume.count > 0 THEN**

**FOR k IN v\_nume.first..v\_nume.last**

**LOOP**

**dbms\_output.put\_line(v\_nume(k));**

**END LOOP;**

**ELSE**

**dbms\_output.put\_line('Nu exista!');**

**END IF;**

**dbms\_output.put\_line('---------------------------------');**

**dbms\_output.put\_line('---------------------------------');**

**END LOOP;**

**END LOOP;**

**END;**

**/**

**BEGIN**

**ex6('MSI');**

**END;**

**/**

**A screenshot of a computer

Description automatically generated**

**A screenshot of a computer

Description automatically generated**

**NU ESTE TOT OUTPUTUL**

|  |
| --- |
| EX7 |

**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.**

***-- Cerința 7:***

***-- Pentru jucatorii care apartin regiunii KOR, afisati campionii pe care ii joaca si folosesc ca resursa mana.***

**CREATE OR REPLACE PROCEDURE ex7(atr\_regiune\_nume regiune.nume%TYPE, atr\_resursa campion.resursa%TYPE)**

**AS**

**CURSOR**

**regiuni(regiune\_nume regiune.nume%TYPE) IS**

**SELECT r.id**

**FROM regiune r**

**WHERE UPPER(r.nume) LIKE UPPER(atr\_regiune\_nume);**

**CURSOR**

**jucatori IS**

**SELECT j.nume, j.id, j.id\_regiune**

**FROM jucator j;**

**CURSOR**

**campioni(jucator\_id jucator.id%TYPE) IS**

**SELECT c.nume || ' ' || '->' || ' ' || c.resursa AS result**

**FROM campion c,**

**campion\_pool cp**

**WHERE c.id = cp.id\_campion**

**AND cp.id\_jucator = jucator\_id**

**AND UPPER(c.resursa) LIKE UPPER(atr\_resursa);**

**v\_nume\_jucator jucator.nume%TYPE;**

**v\_id\_jucator jucator.id%TYPE;**

**v\_id\_regiune regiune.id%TYPE;**

**BEGIN**

**FOR regiune IN regiuni(atr\_regiune\_nume)**

**LOOP**

**dbms\_output.put\_line('Jucatorii din regiunea ' || atr\_regiune\_nume || ' sunt: ');**

**dbms\_output.put\_line('------------------------------------------------');**

**OPEN jucatori;**

**LOOP**

**FETCH jucatori INTO v\_nume\_jucator, v\_id\_jucator, v\_id\_regiune;**

**EXIT WHEN jucatori%NOTFOUND;**

**IF v\_id\_regiune = regiune.id THEN**

**dbms\_output.put\_line('Jucatorul: ' || v\_nume\_jucator);**

**FOR campion IN campioni(v\_id\_jucator)**

**LOOP**

**dbms\_output.put\_line(campion.result);**

**END LOOP;**

**dbms\_output.put\_line('------------------------------------------------');**

**END IF;**

**END LOOP;**

**END LOOP;**

**END;**

**/**

**BEGIN**

**ex7('KOR', 'mana');**

**END;**

**/**

**A screenshot of a computer

Description automatically generated with medium confidence**

**A screenshot of a computer

Description automatically generated with medium confidence**

|  |
| --- |
| EX8 |

**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 tratate.**

***-- Cerința 8:***

***-- Pentru un jucator dat afisati numarul de campioni pe care acesta ii joaca,***

***-- daca acesta nu joaca pe lane-ul mid.***

**CREATE OR REPLACE FUNCTION ex8(atr\_jucator\_id jucator.id%TYPE)**

**RETURN number**

**IS**

**v\_numar\_campioni NUMBER(10);**

**TYPE tip\_tabel IS TABLE OF campion\_pool%ROWTYPE INDEX BY PLS\_INTEGER;**

**v\_tabel tip\_tabel;**

**TYPE tip\_tabel2 IS TABLE OF jucator%ROWTYPE INDEX BY PLS\_INTEGER;**

**v\_tabel2 tip\_tabel2;**

**v\_nume\_jucator jucator.nume%TYPE;**

**negative\_id EXCEPTION;**

**no\_data\_found\_1 EXCEPTION;**

**no\_data\_found\_2 EXCEPTION;**

**no\_player\_found EXCEPTION;**

**BEGIN**

**IF atr\_jucator\_id < 0 THEN**

**RAISE negative\_id;**

**END IF;**

**SELECT \* BULK COLLECT**

**INTO v\_tabel2**

**FROM jucator**

**WHERE id = atr\_jucator\_id;**

**IF SQL%NOTFOUND THEN**

**RAISE no\_player\_found;**

**END IF;**

**SELECT \* BULK COLLECT**

**INTO v\_tabel**

**FROM campion\_pool cp**

**WHERE cp.id\_jucator = atr\_jucator\_id;**

**IF SQL%NOTFOUND THEN**

**RAISE no\_data\_found\_1;**

**END IF;**

**SELECT j.nume INTO v\_nume\_jucator FROM jucator j WHERE j.id = atr\_jucator\_id;**

**SELECT COUNT(c.id)**

**INTO v\_numar\_campioni**

**FROM campion c**

**JOIN campion\_pool cp ON c.id = cp.id\_campion**

**JOIN jucator j ON j.id = cp.id\_jucator**

**WHERE cp.id\_jucator = atr\_jucator\_id**

**AND j.lane NOT LIKE 'MID';**

**IF v\_numar\_campioni = 0 THEN**

**RAISE no\_data\_found\_2;**

**ELSE**

**RETURN v\_numar\_campioni;**

**END IF;**

**EXCEPTION**

**WHEN negative\_id THEN**

**dbms\_output.put\_line('ID-ul nu poate fi negativ!');**

**RETURN -1;**

**WHEN no\_data\_found\_1 THEN**

**dbms\_output.put\_line('Nu exista date pentru acest jucator!');**

**RETURN -1;**

**WHEN no\_data\_found\_2 THEN**

**dbms\_output.put\_line('Jucatorul ' || v\_nume\_jucator || ' joaca pe lane-ul MID!');**

**RETURN -1;**

**WHEN no\_player\_found THEN**

**dbms\_output.put\_line('Nu exista jucator cu acest ID!');**

**RETURN -1;**

**WHEN OTHERS THEN**

**dbms\_output.put\_line('Cod eroare' || SQLCODE);**

**dbms\_output.put\_line('Mesaj eroare' || SQLERRM);**

**RETURN -1;**

**END;**

**/**

**DECLARE**

**v\_numar NUMBER;**

**v\_id\_jucator jucator.id%TYPE := &id\_jucator;**

**BEGIN**

**v\_numar := ex8(v\_id\_jucator);**

**IF v\_numar > -1 THEN**

**dbms\_output.put\_line('Jucatorul cu id-ul ' || v\_id\_jucator || ' joaca ' || v\_numar || ' campioni');**

**END IF;**

**END;**

**/**

***-- Apelare cu id-ul 1 -> 3 campioni***

***-- Apelare cu id-ul 4 -> 2 campioni***

***-- Apelare cu id-ul -1 -> ID-ul nu poate fi negativ!***

***-- Apelare cu id-ul 3 -> Jucatorul Faker joaca pe lane-ul MID!***

***-- Apelare cu id-ul 60 -> Nu exista jucator cu acest ID!***

**DELETE**

**FROM campion\_pool**

**WHERE id\_jucator = 6;**

***-- Apelare cu id-ul 6 -> Nu exista date pentru acest jucator!***

**APELARE CU ID 1**

**Graphical user interface, text

Description automatically generated with medium confidence**

**APELARE CU ID 4**

**A picture containing text, sign

Description automatically generated**

**APELARE CU ID -1**

**Graphical user interface, text

Description automatically generated**

**APELARE CU ID 3**

**A picture containing text, sign

Description automatically generated**

**APELARE CU ID 60**

**A picture containing text, sign

Description automatically generated**

**APELARE CU ID 6**

***DELETE  
FROM* campion\_pool  
*WHERE* id\_jucator = 6;**

**A picture containing text, sign

Description automatically generated**

|  |
| --- |
| EX9 |

**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.**

***-- Cerința 9:***

***-- Pentru o echipa al carei nume este dat, afisati echipele adverse din meciurile ce au loc pe scena “Horn Stage”,***

***-- prezentate de “Jordan Corby”.***

**CREATE OR REPLACE PROCEDURE ex9(nume\_echipa echipa.nume%TYPE)**

**AS**

**TYPE ffic\_index\_echipa IS TABLE OF echipa%ROWTYPE INDEX BY PLS\_INTEGER;**

**v\_echipa ffic\_index\_echipa;**

**TYPE ffic\_index\_echipa\_adversa IS TABLE OF echipa.nume%TYPE INDEX BY PLS\_INTEGER;**

**v\_echipa\_adversa ffic\_index\_echipa\_adversa;**

**v\_id\_echipa echipa.id%TYPE;**

**no\_data\_found\_1 EXCEPTION;**

**no\_data\_found\_2 EXCEPTION;**

**too\_many\_rows EXCEPTION;**

**BEGIN**

**SELECT \* BULK COLLECT**

**INTO v\_echipa**

**FROM echipa e**

**WHERE e.nume = nume\_echipa;**

**IF SQL%NOTFOUND THEN**

**RAISE no\_data\_found\_1;**

**END IF;**

**IF v\_echipa.count > 1 THEN**

**RAISE too\_many\_rows;**

**END IF;**

**SELECT e.id INTO v\_id\_echipa FROM echipa e WHERE e.nume = nume\_echipa;**

**SELECT e.nume BULK COLLECT**

**INTO v\_echipa\_adversa**

**FROM echipa e**

**JOIN meci\_map mm ON e.id = mm.id\_echipa**

**JOIN meci m ON m.id = mm.id\_meci**

**JOIN meci\_header mh ON mh.id\_meci = m.id**

**JOIN scena s ON s.id = mh.id\_scena**

**JOIN prezentator p ON p.id = mh.id\_prezentator**

**WHERE e.nume != nume\_echipa**

**AND s.nume = ‘Horn Stage’**

**AND p.nume = ‘Jordan Corby’**

**AND v\_id\_echipa IN (SELECT mm.id\_echipa FROM meci\_map mm WHERE mm.id\_meci = m.id);**

**IF v\_echipa\_adversa.count = 0 THEN**

**RAISE no\_data\_found\_2;**

**END IF;**

**dbms\_output.put\_line(‘Echipele adversare ale celor din ‘ || nume\_echipa || ‘ sunt: ‘);**

**FOR I IN 1 .. v\_echipa\_adversa.count**

**LOOP**

**dbms\_output.put\_line(v\_echipa\_adversa(i));**

**END LOOP;**

**EXCEPTION**

**WHEN no\_data\_found\_1 THEN**

**dbms\_output.put\_line(‘Nu exista date pentru aceasta echipa!’);**

**WHEN no\_data\_found\_2 THEN**

**dbms\_output.put\_line(‘Nu exista echipe adversare care sa indeplineasca conditiile!’);**

**WHEN too\_many\_rows THEN**

**dbms\_output.put\_line(‘Exista mai multe echipe cu acest nume!’);**

**WHEN OTHERS THEN**

**dbms\_output.put\_line(‘Cod eroare’ || SQLCODE);**

**dbms\_output.put\_line(‘Mesaj eroare’ || SQLERRM);**

**END;**

**/**

**DECLARE**

**nume echipa.nume%TYPE := ‘SKT-T1’;**

**BEGIN**

**ex9(nume);**

**END;**

**/**

***-- Pentru nume = ‘Canids Kalunga’ -> PSG Talon Esports***

***-- Pentru nume = ‘PSG Talon Esports’ -> Canids Kalunga***

***-- Pentru nume = ‘Fnatic’ -> “Nu exista date pentru aceasta echipa!”***

***-- Pentru nume = ‘Order’ -> “Nu exista echipe adversare care sa indeplineasca conditiile!”***

**INSERT INTO echipa(id\_regiune, nume, tag)**

**VALUES (2, ‘SKT-T1’, ‘SKT\_V2’)**

***-- Pentru nume = ‘SKT-T1’ -> “Exista mai multe echipe cu acest nume!”***

**PENTRU NUME ‘Canids Kalunga’**

**Text

Description automatically generated with medium confidence**

**PENTRU NUME ‘PSG Talon Esports’**

**Text

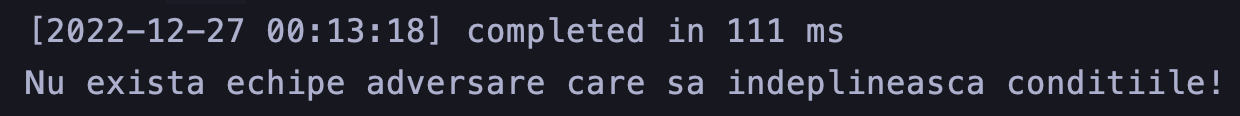
Description automatically generated with medium confidence**

**PENTRU NUME ‘Fnatic’**

**A picture containing text, sign

Description automatically generated**

**PENTRU NUME ‘Order’**

****

**PENTRU NUME ‘SKT-T1’**

***INSERT INTO* echipa(id\_regiune, nume, *tag*)  
*VALUES* (2, 'SKT-T1', 'SKT\_V2')**

**A picture containing text, sign

Description automatically generated**

|  |
| --- |
| EX10 |

**DefinițI un *trigger* de tip LMD la nivel de comandă. DeclanșațI *trigger*-ul.**

***-- Cerința 10:***

***-- Voi realiza un trigger care se va declansa atunci cand vreau sa modific tabela TURNEU cu conditiile:***

***-- 1. Nu pot modifica tabela TURNEU intre orele 08:00-10:00***

***-- 2. Nu pot modifica tabela in ziua de Craciun si Ziua de Anul Nou***

**CREATE OR**

**REPLACE TRIGGER ex10**

**BEFORE**

**INSERT OR UPDATE OR**

**DELETE**

**ON turneu**

**DECLARE**

**BEGIN**

**IF ((TO\_CHAR(SYSDATE, ‘HH24’) BETWEEN 8 AND 10) OR**

**(TO\_CHAR(SYSDATE, ‘MMDD’) = ‘1225’ OR TO\_CHAR(SYSDATE, ‘MMDD’) = ‘0101’)) THEN**

**RAISE\_APPLICATION\_ERROR(-20001, ‘Nu poti modifica tabela in acest interval de timp sau in aceasta zi!’);**

**END IF;**

**END;**

**/**

**INSERT INTO turneu (nume, oras, data)**

**VALUES (‘TurneuName’, ‘Stockholm’, TO\_DATE(‘2022-05-10’, ‘YYYY-MM-DD’));**

**Text

Description automatically generated**

**Pentru a testa trigger-ul am adaugat ca data in care rulez acest exemplu sa fie de asemenea interzisa.**

|  |
| --- |
| EX11 |

**DefinițI un *trigger* de tip LMD la nivel de linie. DeclanșațI *trigger*-ul.**

***-- Cerința 11:***

***-- Voi realiza un trigger care se va declansa atunci cand vreau sa modific tabela meci cu conditiile:***

***-- 1. Data meciului nu poate fi mai veche de 1 Aprilie 2011***

***-- (data in care a avut loc primul joc de League of Legends oficial).***

***-- 2. Data meciului nu poate fi in ziua de Craciun si Ziua de Anul Nou.***

**CREATE**

**OR**

**REPLACE TRIGGER ex11**

**BEFORE**

**INSERT OR UPDATE OR**

**DELETE**

**ON meci**

**FOR EACH ROW**

**DECLARE**

**BEGIN**

**IF (TO\_DATE(:new.data, ‘YYYY-MM-DD’) <= TO\_DATE(‘2011-04-01’, ‘YYYY-MM-DD’) OR**

**(TO\_CHAR(:new.data, ‘MMDD’) = ‘1225’ OR TO\_CHAR(:new.data, ‘MMDD’) = ‘0101’)) THEN**

**RAISE\_APPLICATION\_ERROR(-20001,**

**‘Probleme cu data introdusa,’ ||**

**‘ data nu poate fi mai veche de 1 Aprilie 2011’ ||**

**‘ sau in ziua de Craciun sau in Ziua de Anul Nou!’);**

**END IF;**

**END;**

**INSERT INTO meci (titlu, data)**

**VALUES (‘MeciName’, TO\_DATE(‘2022-01-01’, ‘YYYY-MM-DD’));**

**INSERT INTO meci (titlu, data)**

**VALUES (‘MeciName’, TO\_DATE(‘2010-10-10’, ‘YYYY-MM-DD’));**

**INSERT INTO meci (titlu, data)**

**VALUES (‘MeciName’, TO\_DATE(‘2022-01-01’, ‘YYYY-MM-DD’));**

**INSERT INTO meci (titlu, data)**

**VALUES (‘MeciName’, TO\_DATE(‘2010-10-10’, ‘YYYY-MM-DD’));**

**Text

Description automatically generated**

**Text

Description automatically generated**

|  |
| --- |
| EX12 |

**DefinițI un *trigger* de tip LDD. DeclanșațI *trigger*-ul.**

***-- Cerința 12:***

***-- Voi realiza un trigger care se va declansa atunci cand se executa operatii LDD.***

***-- Acesta va stoca intr-o noua table numita istoric utilizatorul,***

***-- operatia executata si baza de date in care are loc operatia.***

**CREATE TABLE istoric**

**(**

**id NUMBER(10) GENERATED ALWAYS AS IDENTITY**

***CONSTRAINT* istoric\_pk PRIMARY KEY,**

**utilizator VARCHAR2(50)**

***CONSTRAINT* utilizator\_istoric NOT NULL,**

**operatie VARCHAR2(50)**

***CONSTRAINT* operatie\_istoric NOT NULL,**

**baza\_de\_date VARCHAR2(50)**

***CONSTRAINT* baza\_de\_date\_istoric NOT NULL**

**);**

**SELECT \***

**FROM istoric;**

**CREATE**

**OR**

**REPLACE TRIGGER ex12**

**AFTER CREATE OR ALTER OR DROP**

**ON SCHEMA**

**BEGIN**

**INSERT INTO istoric (utilizator, operatie, baza\_de\_date)**

**VALUES (sys.login\_user, sys.sysevent, sys.database\_name);**

**END;**

**/**

**CREATE**

**TABLE**

**test**

**(**

**id NUMBER(10),**

**nume VARCHAR2(50)**

**);**

**ALTER**

**TABLE**

**test**

**ADD**

**(prenume VARCHAR2(50));**

**ALTER TABLE**

**test**

**DROP COLUMN prenume;**

**DROP**

**TABLE**

**test;**

**SELECT \***

**FROM istoric;**

**Graphical user interface, text

Description automatically generated**

|  |
| --- |
| EX13 |

**Definiți unpachet care sa contina toate obiectele definite in cadrul proiectului.**

**CREATE OR REPLACE PACKAGE proiect\_sgbd\_hutan\_mihai AS**

**PROCEDURE ex6(nume\_turneu turneu.nume%TYPE);**

**PROCEDURE ex7(atr\_regiune\_nume regiune.nume%TYPE, atr\_resursa campion.resursa%TYPE);**

**FUNCTION ex8(atr\_jucator\_id jucator.id%TYPE) RETURN NUMBER;**

**PROCEDURE ex9(nume\_echipa echipa.nume%TYPE);**

**END proiect\_sgbd\_hutan\_mihai;**

**/**

**CREATE OR REPLACE PACKAGE BODY proiect\_sgbd\_hutan\_mihai**

**AS**

***-- Cerința 6:***

***-- Pentru un turneu al carui nume este dat, pentru fiecare regiune, afisati toate echipele si jucatorii ce apartin***

***-- acelei regiuni sau mesajul 'Nu exista!'.***

**PROCEDURE ex6(nume\_turneu turneu.nume%TYPE)**

**AS**

**TYPE tabl\_idx IS TABLE OF regiune%ROWTYPE INDEX BY PLS\_INTEGER;**

**v\_regiuni tabl\_idx;**

**TYPE tip\_lista\_nested IS TABLE OF echipa%ROWTYPE;**

**v\_echipe tip\_lista\_nested := tip\_lista\_nested();**

**TYPE tabl\_index IS TABLE OF VARCHAR2(200) INDEX BY PLS\_INTEGER;**

**v\_nume tabl\_index;**

**v\_numar NUMBER(5);**

**BEGIN**

**SELECT \* BULK COLLECT INTO v\_regiuni FROM regiune;**

**SELECT COUNT(\*)**

**INTO v\_numar**

**FROM echipa e,**

**turneu t,**

**grupa\_map gm,**

**grupa g**

**WHERE e.id = gm.id\_echipa**

**AND t.id = gm.id\_grupa**

**AND g.id = gm.id\_grupa**

**AND t.id = g.id\_turneu**

**AND UPPER(t.nume) LIKE UPPER(nume\_turneu);**

**v\_echipe.extend(v\_numar + 1);**

**SELECT e.id, e.nume, e.tag, e.id\_regiune BULK COLLECT**

**INTO v\_echipe**

**FROM echipa e,**

**turneu t,**

**grupa\_map gm,**

**grupa g**

**WHERE e.id = gm.id\_echipa**

**AND t.id = gm.id\_grupa**

**AND g.id = gm.id\_grupa**

**AND t.id = g.id\_turneu**

**AND UPPER(t.nume) LIKE UPPER(nume\_turneu);**

**FOR i IN v\_regiuni.first..v\_regiuni.last**

**LOOP**

**dbms\_output.put\_line('REGIUNEA: ' || v\_regiuni(i).nume);**

**dbms\_output.put\_line('---------------------------------');**

**FOR j IN v\_echipe.first..v\_echipe.last**

**LOOP**

**dbms\_output.put\_line('ECHIPA: ' || v\_echipe(j).nume);**

**dbms\_output.put\_line('---------------------------------');**

**SELECT j.nume BULK COLLECT**

**INTO v\_nume**

**FROM jucator j,**

**echipa e,**

**regiune r**

**WHERE j.id\_echipa = e.id**

**AND j.id\_regiune = r.id**

**AND e.nume = v\_echipe(j).nume**

**AND r.nume = v\_regiuni(i).nume;**

**IF v\_nume.count > 0 THEN**

**FOR k IN v\_nume.first..v\_nume.last**

**LOOP**

**dbms\_output.put\_line(v\_nume(k));**

**END LOOP;**

**ELSE**

**dbms\_output.put\_line('Nu exista!');**

**END IF;**

**dbms\_output.put\_line('---------------------------------');**

**dbms\_output.put\_line('---------------------------------');**

**END LOOP;**

**END LOOP;**

**END ex6;**

***-- Cerința 7:***

***-- Pentru jucatorii care apartin regiunii KOR, afisati campionii pe care ii joaca si folosesc ca resursa mana.***

**PROCEDURE ex7(atr\_regiune\_nume regiune.nume%TYPE, atr\_resursa campion.resursa%TYPE)**

**AS**

**CURSOR**

**regiuni(regiune\_nume regiune.nume%TYPE) IS**

**SELECT r.id**

**FROM regiune r**

**WHERE UPPER(r.nume) LIKE UPPER(atr\_regiune\_nume);**

**CURSOR**

**jucatori IS**

**SELECT j.nume, j.id, j.id\_regiune**

**FROM jucator j;**

**CURSOR**

**campioni(jucator\_id jucator.id%TYPE) IS**

**SELECT c.nume || ' ' || '->' || ' ' || c.resursa AS result**

**FROM campion c,**

**campion\_pool cp**

**WHERE c.id = cp.id\_campion**

**AND cp.id\_jucator = jucator\_id**

**AND UPPER(c.resursa) LIKE UPPER(atr\_resursa);**

**v\_nume\_jucator jucator.nume%TYPE;**

**v\_id\_jucator jucator.id%TYPE;**

**v\_id\_regiune regiune.id%TYPE;**

**BEGIN**

**FOR regiune IN regiuni(atr\_regiune\_nume)**

**LOOP**

**dbms\_output.put\_line('Jucatorii din regiunea ' || atr\_regiune\_nume || ' sunt: ');**

**dbms\_output.put\_line('------------------------------------------------');**

**OPEN jucatori;**

**LOOP**

**FETCH jucatori INTO v\_nume\_jucator, v\_id\_jucator, v\_id\_regiune;**

**EXIT WHEN jucatori%NOTFOUND;**

**IF v\_id\_regiune = regiune.id THEN**

**dbms\_output.put\_line('Jucatorul: ' || v\_nume\_jucator);**

**FOR campion IN campioni(v\_id\_jucator)**

**LOOP**

**dbms\_output.put\_line(campion.result);**

**END LOOP;**

**dbms\_output.put\_line('------------------------------------------------');**

**END IF;**

**END LOOP;**

**END LOOP;**

**END ex7;**

***-- Cerința 8:***

***-- Pentru un jucator dat afisati numarul de campioni pe care acesta ii joaca,***

***-- daca acesta nu joaca pe lane-ul mid.***

**FUNCTION ex8(atr\_jucator\_id jucator.id%TYPE)**

**RETURN number**

**IS**

**v\_numar\_campioni NUMBER(10);**

**TYPE tip\_tabel IS TABLE OF campion\_pool%ROWTYPE INDEX BY PLS\_INTEGER;**

**v\_tabel tip\_tabel;**

**TYPE tip\_tabel2 IS TABLE OF jucator%ROWTYPE INDEX BY PLS\_INTEGER;**

**v\_tabel2 tip\_tabel2;**

**v\_nume\_jucator jucator.nume%TYPE;**

**negative\_id EXCEPTION;**

**no\_data\_found\_1 EXCEPTION;**

**no\_data\_found\_2 EXCEPTION;**

**no\_player\_found EXCEPTION;**

**BEGIN**

**IF atr\_jucator\_id < 0 THEN**

**RAISE negative\_id;**

**END IF;**

**SELECT \* BULK COLLECT**

**INTO v\_tabel2**

**FROM jucator**

**WHERE id = atr\_jucator\_id;**

**IF SQL%NOTFOUND THEN**

**RAISE no\_player\_found;**

**END IF;**

**SELECT \* BULK COLLECT**

**INTO v\_tabel**

**FROM campion\_pool cp**

**WHERE cp.id\_jucator = atr\_jucator\_id;**

**IF SQL%NOTFOUND THEN**

**RAISE no\_data\_found\_1;**

**END IF;**

**SELECT j.nume INTO v\_nume\_jucator FROM jucator j WHERE j.id = atr\_jucator\_id;**

**SELECT COUNT(c.id)**

**INTO v\_numar\_campioni**

**FROM campion c**

**JOIN campion\_pool cp ON c.id = cp.id\_campion**

**JOIN jucator j ON j.id = cp.id\_jucator**

**WHERE cp.id\_jucator = atr\_jucator\_id**

**AND j.lane NOT LIKE 'MID';**

**IF v\_numar\_campioni = 0 THEN**

**RAISE no\_data\_found\_2;**

**ELSE**

**RETURN v\_numar\_campioni;**

**END IF;**

**EXCEPTION**

**WHEN negative\_id THEN**

**dbms\_output.put\_line('ID-ul nu poate fi negativ!');**

**RETURN -1;**

**WHEN no\_data\_found\_1 THEN**

**dbms\_output.put\_line('Nu exista date pentru acest jucator!');**

**RETURN -1;**

**WHEN no\_data\_found\_2 THEN**

**dbms\_output.put\_line('Jucatorul ' || v\_nume\_jucator || ' joaca pe lane-ul MID!');**

**RETURN -1;**

**WHEN no\_player\_found THEN**

**dbms\_output.put\_line('Nu exista jucator cu acest ID!');**

**RETURN -1;**

**WHEN OTHERS THEN**

**dbms\_output.put\_line('Cod eroare' || SQLCODE);**

**dbms\_output.put\_line('Mesaj eroare' || SQLERRM);**

**RETURN -1;**

**END ex8;**

***-- Cerința 9:***

***-- Pentru o echipa al carei nume este dat, afisati echipele adverse din meciurile ce au loc pe scena "Horn Stage",***

***-- prezentate de "Jordan Corby".***

**PROCEDURE ex9(nume\_echipa echipa.nume%TYPE)**

**AS**

**TYPE tabel\_index\_echipa IS TABLE OF echipa%ROWTYPE INDEX BY PLS\_INTEGER;**

**v\_echipa tabel\_index\_echipa;**

**TYPE tabel\_index\_echipa\_adversa IS TABLE OF echipa.nume%TYPE INDEX BY PLS\_INTEGER;**

**v\_echipa\_adversa tabel\_index\_echipa\_adversa;**

**v\_id\_echipa echipa.id%TYPE;**

**no\_data\_found\_1 EXCEPTION;**

**no\_data\_found\_2 EXCEPTION;**

**too\_many\_rows EXCEPTION;**

**BEGIN**

**SELECT \* BULK COLLECT**

**INTO v\_echipa**

**FROM echipa e**

**WHERE e.nume = nume\_echipa;**

**IF SQL%NOTFOUND THEN**

**RAISE no\_data\_found\_1;**

**END IF;**

**IF v\_echipa.count > 1 THEN**

**RAISE too\_many\_rows;**

**END IF;**

**SELECT e.id INTO v\_id\_echipa FROM echipa e WHERE e.nume = nume\_echipa;**

**SELECT e.nume BULK COLLECT**

**INTO v\_echipa\_adversa**

**FROM echipa e**

**JOIN meci\_map mm ON e.id = mm.id\_echipa**

**JOIN meci m ON m.id = mm.id\_meci**

**JOIN meci\_header mh ON mh.id\_meci = m.id**

**JOIN scena s ON s.id = mh.id\_scena**

**JOIN prezentator p ON p.id = mh.id\_prezentator**

**WHERE e.nume != nume\_echipa**

**AND s.nume = 'Horn Stage'**

**AND p.nume = 'Jordan Corby'**

**AND v\_id\_echipa IN (SELECT mm.id\_echipa FROM meci\_map mm WHERE mm.id\_meci = m.id);**

**IF v\_echipa\_adversa.count = 0 THEN**

**RAISE no\_data\_found\_2;**

**END IF;**

**dbms\_output.put\_line('Echipele adversare ale celor din ' || nume\_echipa || ' sunt: ');**

**FOR i IN 1 .. v\_echipa\_adversa.count**

**LOOP**

**dbms\_output.put\_line(v\_echipa\_adversa(i));**

**END LOOP;**

**EXCEPTION**

**WHEN no\_data\_found\_1 THEN**

**dbms\_output.put\_line('Nu exista date pentru aceasta echipa!');**

**WHEN no\_data\_found\_2 THEN**

**dbms\_output.put\_line('Nu exista echipe adversare care sa indeplineasca conditiile!');**

**WHEN too\_many\_rows THEN**

**dbms\_output.put\_line('Exista mai multe echipe cu acest nume!');**

**WHEN OTHERS THEN**

**dbms\_output.put\_line('Cod eroare' || SQLCODE);**

**dbms\_output.put\_line('Mesaj eroare' || SQLERRM);**

**END ex9;**

**END proiect\_sgbd\_hutan\_mihai;**

**/**

***-- TESTARE PACKAGE***

**BEGIN**

**proiect\_sgbd\_hutan\_mihai.ex6('MSI');**

**proiect\_sgbd\_hutan\_mihai.ex7('KOR', 'mana');**

***-- Apelare cu id-ul 1 -> 3 campioni***

**dbms\_output.put\_line('Rezultat ex8: ' || proiect\_sgbd\_hutan\_mihai.ex8(1));**

***-- Apelare cu id-ul 4 -> 2 campioni***

**dbms\_output.put\_line('Rezultat ex8: ' || proiect\_sgbd\_hutan\_mihai.ex8(4));**

***-- Apelare cu id-ul -1 -> ID-ul nu poate fi negativ!***

**dbms\_output.put\_line('Rezultat ex8: ' || proiect\_sgbd\_hutan\_mihai.ex8(-1));**

***-- Apelare cu id-ul 3 -> Jucatorul Faker joaca pe lane-ul MID!***

**dbms\_output.put\_line('Rezultat ex8: ' || proiect\_sgbd\_hutan\_mihai.ex8(3));**

***-- Apelare cu id-ul 60 -> Nu exista jucator cu acest ID!***

**dbms\_output.put\_line('Rezultat ex8: ' || proiect\_sgbd\_hutan\_mihai.ex8(60));**

**END;**

**/**

**DELETE**

**FROM campion\_pool**

**WHERE id\_jucator = 6;**

**BEGIN**

***-- Apelare cu id-ul 6 -> Nu exista date pentru acest jucator!***

**dbms\_output.put\_line('Rezultat ex8: ' || proiect\_sgbd\_hutan\_mihai.ex8(6));**

**END;**

**/**

**BEGIN**

***-- Pentru nume = 'Canids Kalunga' -> PSG Talon Esports***

**proiect\_sgbd\_hutan\_mihai.ex9('Canids Kalunga');**

***-- Pentru nume = 'PSG Talon Esports' -> Canids Kalunga***

**proiect\_sgbd\_hutan\_mihai.ex9('PSG Talon Esports');**

***-- Pentru nume = 'Fnatic' -> "Nu exista date pentru aceasta echipa!"***

**proiect\_sgbd\_hutan\_mihai.ex9('Fnatic');**

***-- Pentru nume = 'Order' -> "Nu exista echipe adversare care sa indeplineasca conditiile!"***

**proiect\_sgbd\_hutan\_mihai.ex9('Order');**

**END;**

**/**

**INSERT INTO echipa(id\_regiune, nume, tag)**

**VALUES (2, 'SKT-T1', 'SKT\_V2');**

**BEGIN**

***-- Pentru nume = 'SKT-T1' -> "Exista mai multe echipe cu acest nume!"***

**proiect\_sgbd\_hutan\_mihai.ex9('SKT-T1');**

**END;**

**/**

**LA RULAREA TESTELOR PENTRU PACKAGE VOM OBTINE ACELEASI REZULTATE CA SI IN CERINTELE IMPLEMENTATE IN AFARA PACKAGE-ULUI DEOARECE AM TESTAT ACELEASI CAZURI.**