Gestionare raiduri în jocuri MMORPG

Clinciu Diana-Claudia

Grupa 244

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

Din cauza volumului mare de jucători care există de obicei într-un joc de rol online în masă (MMORPG) este necesară crearea unei baze de date care să rețină informații despre jucători și activitatea lor, raiduri, inamici etc.

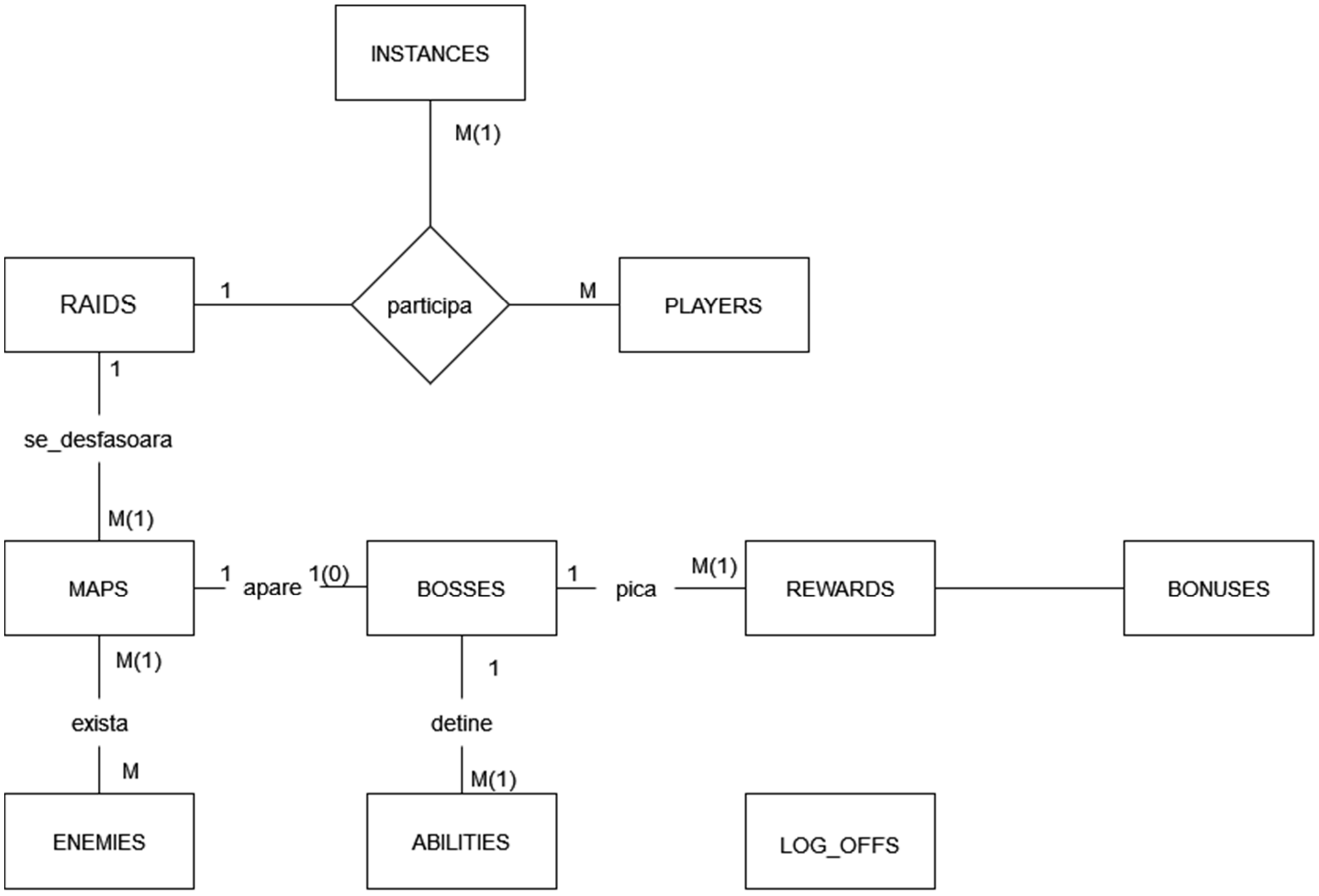
La un raid pot participa mai mulți jucători. Jucătorii participă la un raid printr-o instanță. Pentru fiecare jucător se reține: numele, *realm*-ul din care vine, rasa, genul, clasa, rolul și nivelul. Pentru fiecare raid se reține: numele, locația, numărul maxim de jucători și nivelul minim necesar pe care un jucător trebuie să îl aibă pentru a lua parte la raid. Pentru o instanță se va reține data creării.

Un raid se poate desfășura pe mai multe hărți. Pentru o hartă se va reține numele hărții. Pe o hartă se află mai mulți inamici pe care jucătorii trebuie să îi înfrângă că să ajungă la *boss*. Pe unele hărți se află și *boss-i*. Pe o hartă se poate afla doar un singur *boss*. Pot exista hărți care au și *boss-i* și inamici pe ele. Pentru un inamic se va reține: numele, viața și *damage*-ul.

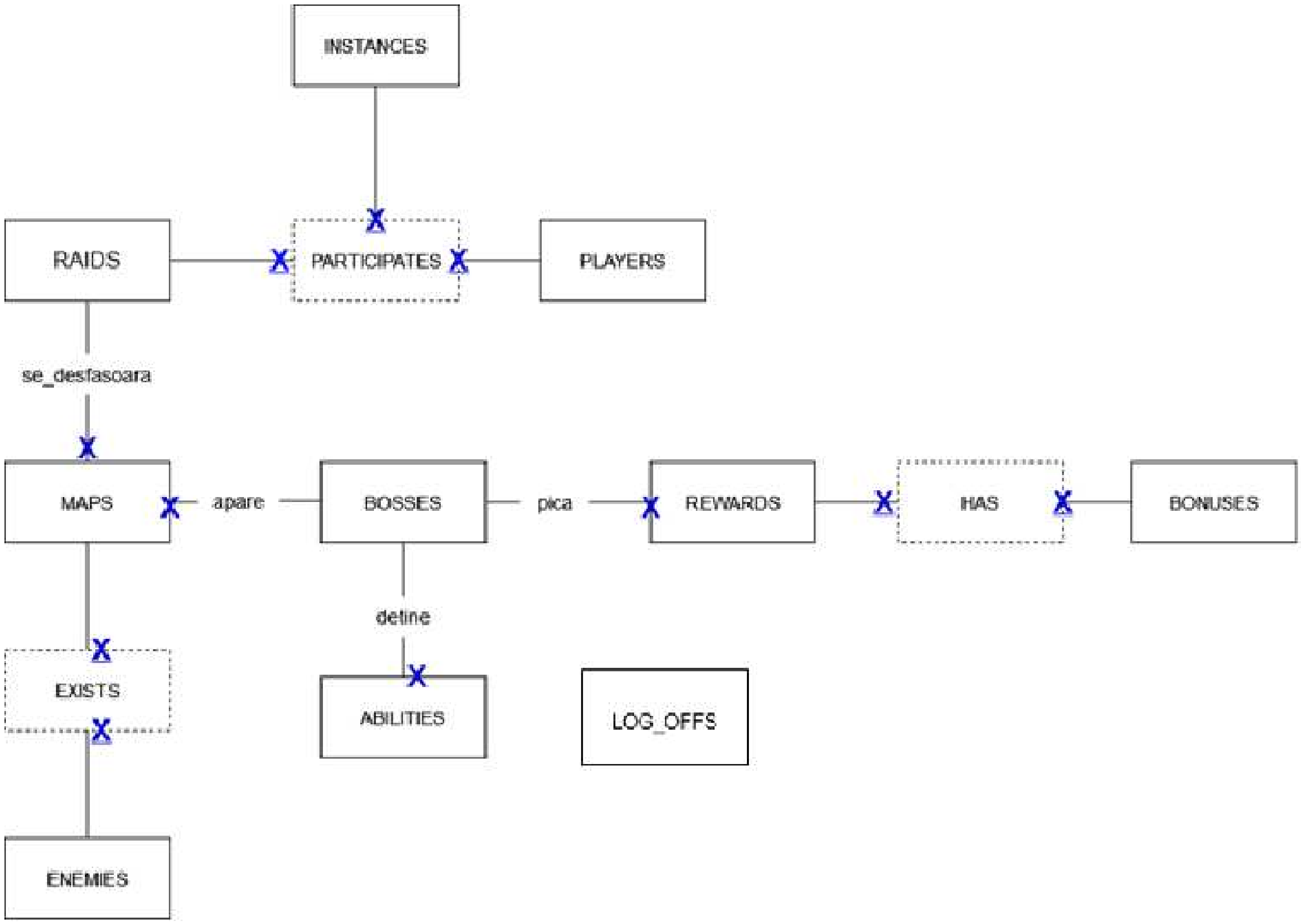
Pentru un *boss* se reține: numele, viata, *damage*-ul și *mana.* Un *boss* are mai multe abilități. Pentru o abilitate se reține: numele, *mana* necesară pentru a folosi abilitatea si *damege*-ul. Atunci când un *boss* este înfrânt jucătorii primesc o recompensă. De la fiecare *boss* pot pica mai multe recompense. Recompensele sunt specifice fiecărui *boss*. Pentru un recompensă se reține: numele, o scurtă descriere, prețul la vânzare și bonusurile. Pentru bonusuri se reține: numele bonusului și un număr, bonusul efectiv (ex: Magic: 20).

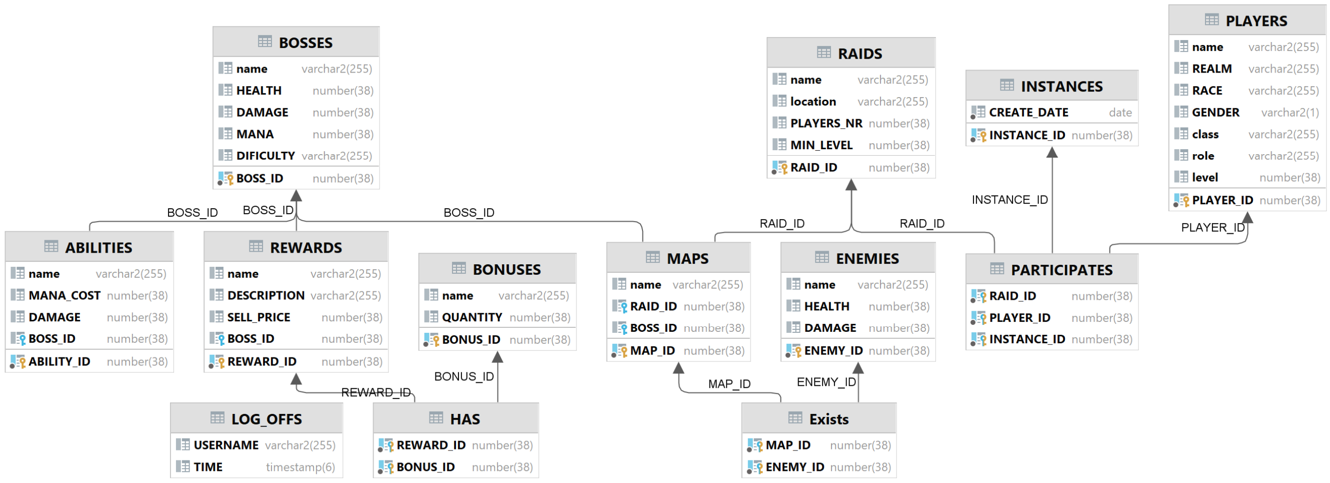
În plus, vom reține informații despre delogările utilizatorilor bazei de date într-o tabelă.

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).

```
CREATE TABLE raids
(
    raid_id    NUMBER(38) PRIMARY KEY,
    "name"     VARCHAR(255),
    "location" VARCHAR(255),
    players_nr NUMBER(38) CHECK (players_nr BETWEEN 5 AND 45),
    min_level  NUMBER(38) CHECK (min_level BETWEEN 10 AND 60)
);

CREATE TABLE players
(
    player_id NUMBER(38) PRIMARY KEY,
    "name"    VARCHAR(255),
    realm     VARCHAR(255),
    race      VARCHAR(255),
    gender    VARCHAR(1) CHECK (gender LIKE 'M' OR gender LIKE 'F'),
    "class"   VARCHAR(255),
    "role"    VARCHAR(255) CHECK ("role" LIKE 'healer' OR "role" LIKE 'tank' OR "role" LIKE
'damage-dealer'),
    "level"   NUMBER(38) DEFAULT 1 CHECK ("level" BETWEEN 1 AND 60)
);


CREATE TABLE instances
(
    instance_id NUMBER(38) PRIMARY KEY,
    create_date DATE NOT NULL
);

CREATE TABLE bosses
(
    boss_id NUMBER(38) PRIMARY KEY,
    "name"  VARCHAR(255),
    health  NUMBER(38),
    damage  NUMBER(38),
    mana    NUMBER(38)
);

ALTER TABLE bosses
    ADD dificulty VARCHAR(255);

CREATE TABLE maps
(
    map_id  NUMBER(38) PRIMARY KEY,
    "name"  VARCHAR(255),
    raid_id NUMBER(38) REFERENCES raids (raid_id),
    boss_id NUMBER(38) REFERENCES bosses (boss_id)
);

CREATE TABLE enemies
(
    enemy_id NUMBER(38) PRIMARY KEY,
    "name"   VARCHAR(255),
    health   NUMBER(38),
    damage   NUMBER(38)
);


CREATE TABLE abilities
(
    ability_id NUMBER(38) PRIMARY KEY,
    "name"     VARCHAR(255),
    mana_cost  NUMBER(38),
    damage     NUMBER(38),
    boss_id    NUMBER(38) REFERENCES bosses (boss_id)
);

CREATE TABLE rewards
(
    reward_id   NUMBER(38) PRIMARY KEY,
    "name"      VARCHAR(255),
    description VARCHAR(255),
    sell_price  NUMBER(38),
    boss_id     NUMBER(38) REFERENCES bosses (boss_id)
);

CREATE TABLE bonuses
(
    bonus_id NUMBER(38) PRIMARY KEY,
    "name"   VARCHAR(255),
    quantity NUMBER(38)
);


CREATE TABLE participates
(
    raid_id     NUMBER(38) REFERENCES raids (raid_id),
    player_id   NUMBER(38) REFERENCES players (player_id),
    instance_id NUMBER(38) REFERENCES instances (instance_id),
    PRIMARY KEY (raid_id, player_id, instance_id)
);
```

```
CREATE TABLE "Exists"
(
    map_id   NUMBER(38) REFERENCES maps (map_id),
    enemy_id NUMBER(38) REFERENCES enemies (enemy_id),
    PRIMARY KEY (map_id, enemy_id)
);

CREATE TABLE has
(
    reward_id NUMBER(38) REFERENCES rewards (reward_id),
    bonus_id  NUMBER(38) REFERENCES bonuses (bonus_id),
    PRIMARY KEY (reward_id, bonus_id)
);

CREATE TABLE log_offs
(
    username VARCHAR(255),
    time     TIMESTAMP
);
```

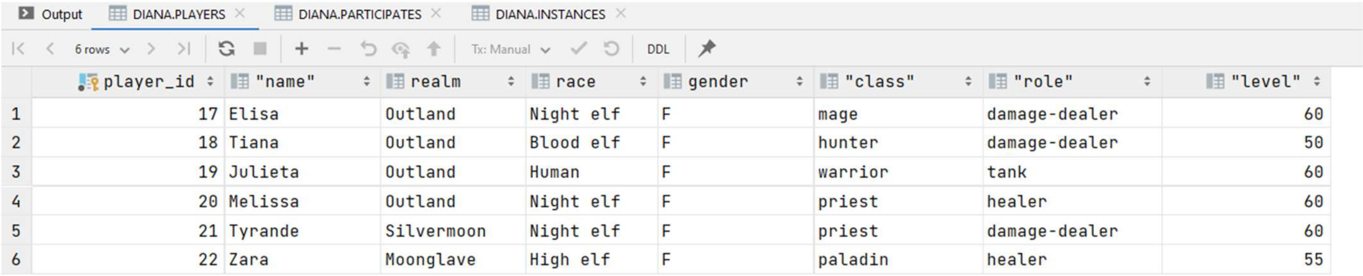
5. Adăugați informații coerente în tabelele create (minim 5 înregistrări pentru fiecare entitate independentă; minim 10 înregistrări pentru tabela asociativă).

```
CREATE SEQUENCE ids
    START WITH 1
    INCREMENT BY 1
    MINVALUE 0
    MAXVALUE 1000000
    NOCYCLE;

INSERT INTO raids
VALUES (ids.nextval, 'Eternal Palace', 'Nazjatar', 5, 60);
INSERT INTO raids
VALUES (ids.nextval, 'Sanctum of Domination', 'Torghast', 5, 40);
INSERT INTO raids
VALUES (ids.nextval, 'Icecrown Citadel', 'Icecrown', 5, 35);
INSERT INTO raids
VALUES (ids.nextval, 'The Nighthold', 'Suramar', 5, 35);
INSERT INTO raids
VALUES (ids.nextval, 'Court of Stars', 'Suramar', 5, 50);



INSERT INTO bosses
VALUES (ids.nextval, 'Queen Azhara', 121000, 4000, 100, NULL);
INSERT INTO bosses
VALUES (ids.nextval, 'Sylvanas Windrunner', 143000, 6000, 400, NULL);
INSERT INTO bosses
VALUES (ids.nextval, 'Lich King', 4300, 300, 40, NULL);
INSERT INTO bosses
VALUES (ids.nextval, 'Elisande', 6300, 700, 60, NULL);
INSERT INTO bosses
VALUES (ids.nextval, 'Talixae Flamewreath', 100000, 3000, 600, NULL);



INSERT INTO maps
VALUES (ids.nextval, 'Queens throne', 1, 6);
INSERT INTO maps (map_id, "name", raid_id)
VALUES (ids.nextval, 'The entrance', 1);
INSERT INTO maps
VALUES (ids.nextval, 'Pinnacle of Domination', 2, 7);
INSERT INTO maps
VALUES (ids.nextval, 'The nightspire', 4, 9);
INSERT INTO maps
VALUES (ids.nextval, 'The Frozen Throne', 3, 8);
INSERT INTO maps
VALUES (ids.nextval, 'Court of Stars', 5, 10);



INSERT INTO players
VALUES (ids.nextval, 'Elisa', 'Outland', 'Night elf', 'F', 'mage', 'damage-dealer', 60);
INSERT INTO players
```

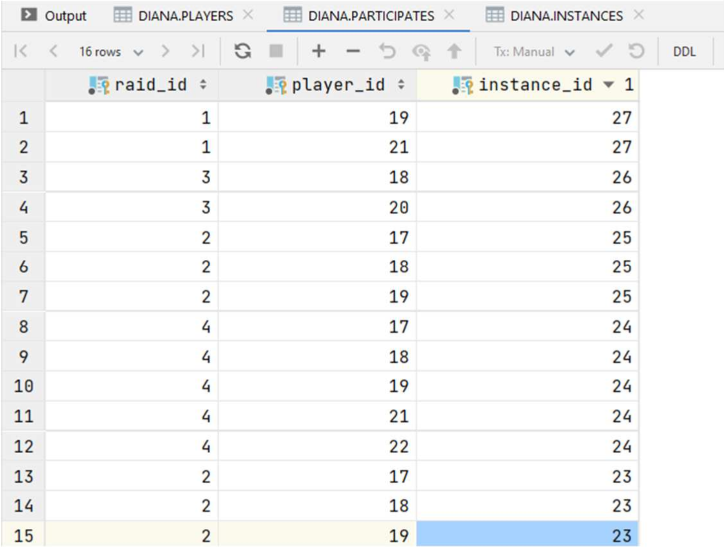
```sql
VALUES (ids.nextval, 'Tiana', 'Outland', 'Blood elf', 'F', 'hunter', 'damage-dealer', 50);
INSERT INTO players
VALUES (ids.nextval, 'Julieta', 'Outland', 'Human', 'F', 'warrior', 'tank', 60);
INSERT INTO players
VALUES (ids.nextval, 'Melissa', 'Outland', 'Night elf', 'F', 'priest', 'healer', 60);
INSERT INTO players
VALUES (ids.nextval, 'Tyrande', 'Silvermoon', 'Night elf', 'F', 'priest', 'damage-dealer',
60);
INSERT INTO players
VALUES (ids.nextval, 'Zara', 'Moonglave', 'High elf', 'F', 'paladin', 'healer', 55);



INSERT INTO instances
VALUES (ids.nextval, DATE'2022-06-10');
INSERT INTO instances
VALUES (ids.nextval, DATE'2022-06-11');
INSERT INTO instances
VALUES (ids.nextval, DATE'2022-06-24');
INSERT INTO instances
VALUES (ids.nextval, DATE'2022-07-24');
INSERT INTO instances
VALUES (ids.nextval, DATE'2022-07-25');


INSERT INTO participates
VALUES (2, 17, 23);
INSERT INTO participates
VALUES (2, 18, 23);
INSERT INTO participates
VALUES (2, 19, 23);
INSERT INTO participates
VALUES (2, 20, 23);
INSERT INTO participates
VALUES (4, 21, 24);
INSERT INTO participates
VALUES (4, 22, 24);
INSERT INTO participates
VALUES (4, 17, 24);
INSERT INTO participates
VALUES (4, 18, 24);
INSERT INTO participates
VALUES (4, 19, 24);
INSERT INTO participates
VALUES (2, 17, 25);
INSERT INTO participates
VALUES (2, 18, 25);
INSERT INTO participates
VALUES (2, 19, 25);
INSERT INTO participates
VALUES (3, 20, 26);
INSERT INTO participates
VALUES (3, 18, 26);
INSERT INTO participates
VALUES (1, 19, 27);
INSERT INTO participates
VALUES (1, 21, 27);


INSERT INTO rewards
VALUES (ids.nextval, 'Bindings of the Herald', 'Binds when picked up', 320, 7);
INSERT INTO rewards
VALUES (ids.nextval, 'The Lady''s Necklace',
        'The necklace shimmers, a remarkable piece of jewelry that doesn''t seem to succumb to
the gloom that permeates the Ghostlands. Given the inscription, and where you found it, it
must be the lost property of Lady Sylvanas Windrunner herself!',
        1000, 7);
INSERT INTO rewards
VALUES (ids.nextval, 'Convergence of Fates', 'Binds when picked up', 400, 9);
INSERT INTO rewards
VALUES (ids.nextval, 'Mana-Tinged Pack',
        'Contains assorted potions and magical reagents from the Broken Isles. Looted by
players with the Manaseeker enchantment.',
        50, 10);
INSERT INTO rewards
VALUES (ids.nextval, 'Warmace of Menethil', 'Binds when picked up', 430, 8);
INSERT INTO rewards
```

```sql
VALUES (ids.nextval, 'Cursed Lover''s Ring', 'No not touch! Ups I''ve touched it', 300, 6);
INSERT INTO rewards (reward_id, "name", description, boss_id)
VALUES (ids.nextval, 'Banshee''s Bow', 'A bow made out of a human spine. It has o dark violet
glow.', 7);



INSERT INTO bonuses
VALUES (ids.nextval, 'Stamina', 18);
INSERT INTO bonuses
VALUES (ids.nextval, 'Haste', 39);
INSERT INTO bonuses
VALUES (ids.nextval, 'Versatility', 29);
INSERT INTO bonuses
VALUES (ids.nextval, 'Magic', 100);
INSERT INTO bonuses
VALUES (ids.nextval, 'Stamina', 50);
INSERT INTO bonuses
VALUES (ids.nextval, 'Atack speed', 20);
INSERT INTO bonuses
VALUES (ids.nextval, 'Mastery', 30);

INSERT INTO has
VALUES (28, 35);
INSERT INTO has
VALUES (28, 36);
INSERT INTO has
VALUES (28, 37);
INSERT INTO has
VALUES (30, 37);
INSERT INTO has
VALUES (31, 37);
INSERT INTO has
VALUES (31, 38);
INSERT INTO has
VALUES (31, 39);
INSERT INTO has
VALUES (32, 40);
INSERT INTO has
VALUES (32, 41);
INSERT INTO has
VALUES (32, 39);
INSERT INTO has
VALUES (29, 39);

INSERT INTO enemies
VALUES (ids.nextval, 'Fallen angel', 100, 12);
INSERT INTO enemies
VALUES (ids.nextval, 'Agatha', 200, 24);
INSERT INTO enemies
VALUES (ids.nextval, 'Fellfire huntdog', 200, 32);
INSERT INTO enemies
VALUES (ids.nextval, 'Shivara', 300, 14);
INSERT INTO enemies
VALUES (ids.nextval, 'Skeleton', 300, 12);


INSERT INTO "Exists"
VALUES (11, 42);
INSERT INTO "Exists"
VALUES (11, 43);
INSERT INTO "Exists"
VALUES (12, 44);
INSERT INTO "Exists"
VALUES (12, 45);
INSERT INTO "Exists"
VALUES (13, 46);
INSERT INTO "Exists"
VALUES (13, 42);
INSERT INTO "Exists"
VALUES (14, 43);
INSERT INTO "Exists"
VALUES (14, 44);
INSERT INTO "Exists"
VALUES (15, 45);
INSERT INTO "Exists"
VALUES (15, 46);
```

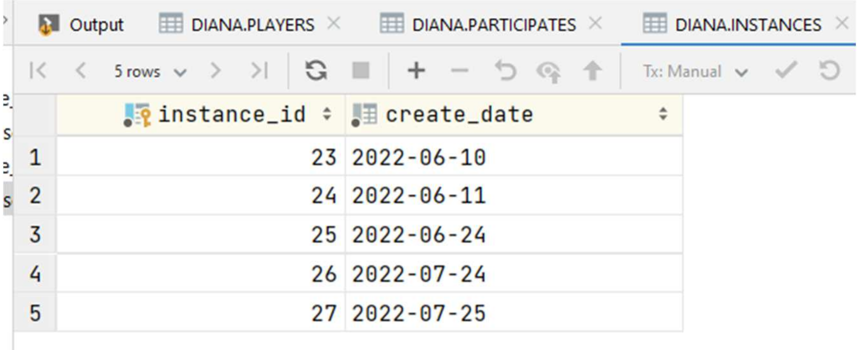
```
INSERT INTO "Exists"
VALUES (16, 44);
INSERT INTO "Exists"
VALUES (16, 45);


INSERT INTO abilities
VALUES (ids.nextval, 'Fury of Frostmourne', 10, 100, 8);
INSERT INTO abilities
VALUES (ids.nextval, 'Raise Dead', 20, 120, 8);
INSERT INTO abilities
VALUES (ids.nextval, 'Domination Arrows', 11, 101, 7);
INSERT INTO abilities
VALUES (ids.nextval, 'Stop Time', 12, 130, 9);
INSERT INTO abilities
VALUES (ids.nextval, 'Bond of Flame', 20, 300, 10);
INSERT INTO abilities
VALUES (ids.nextval, 'Catastrophic Failure', 18, 180, 6);
```



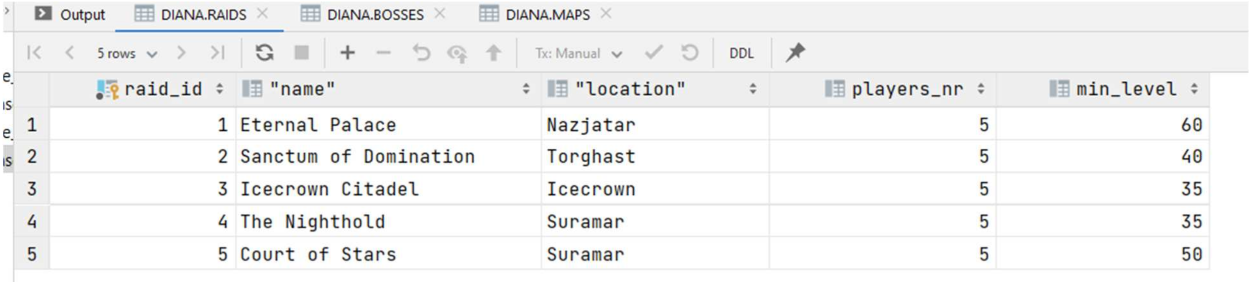
	player_id	"name"	realm	race	gender	"class"	"role"	"level"
1	17	Elisa	Outland	Night elf	F	mage	damage-dealer	60
2	18	Tiana	Outland	Blood elf	F	hunter	damage-dealer	50
3	19	Julieta	Outland	Human	F	warrior	tank	60
4	20	Melissa	Outland	Night elf	F	priest	healer	60
5	21	Tyrande	Silvermoon	Night elf	F	priest	damage-dealer	60
6	22	Zara	Moonglave	High elf	F	paladin	healer	55



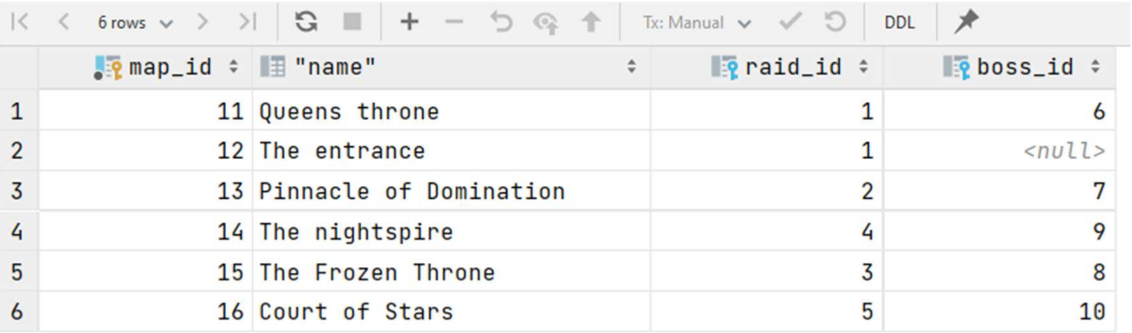
	raid_id	player_id	instance_id 1
1	1	19	27
2	1	21	27
3	3	18	26
4	3	20	26
5	2	17	25
6	2	18	25
7	2	19	25
8	4	17	24
9	4	18	24
10	4	19	24
11	4	21	24
12	4	22	24
13	2	17	23
14	2	18	23
15	2	19	23



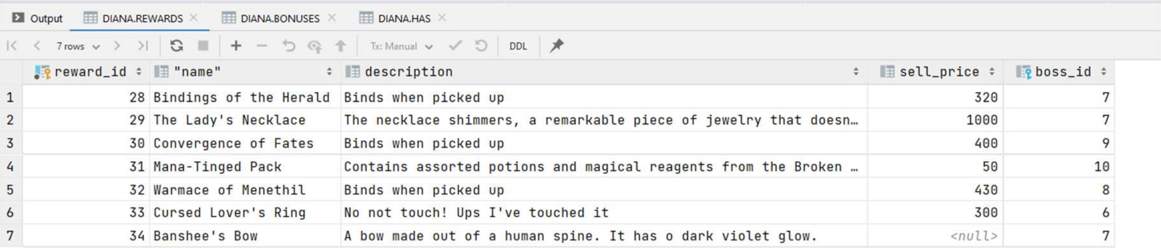
	instance_id	create_date
1	23	2022-06-10
2	24	2022-06-11
3	25	2022-06-24
4	26	2022-07-24
5	27	2022-07-25



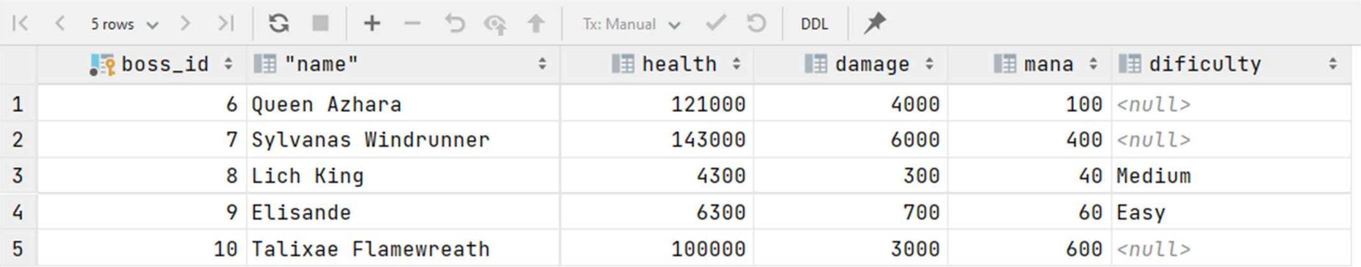
	raid_id	"name"	"location"	players_nr	min_level
1	1	Eternal Palace	Nazjatar	5	60
2	2	Sanctum of Domination	Torghast	5	40
3	3	Icecrown Citadel	Icecrown	5	35
4	4	The Nighthold	Suramar	5	35
5	5	Court of Stars	Suramar	5	50



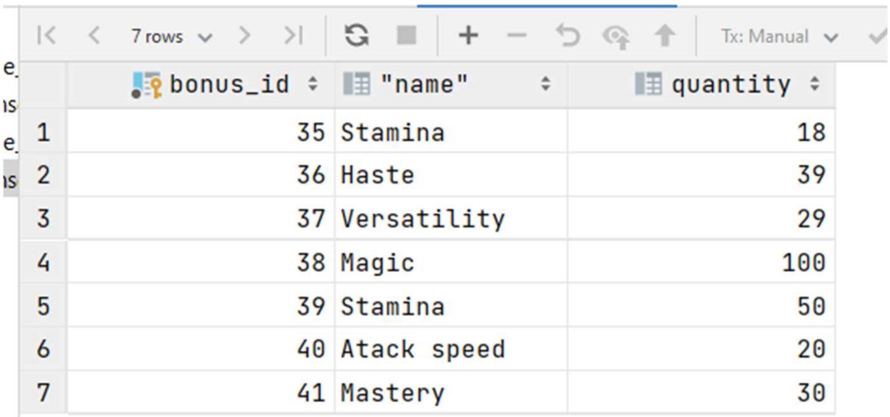
	map_id	"name"	raid_id	boss_id
1	11	Queens throne	1	6
2	12	The entrance	1	<null>
3	13	Pinnacle of Domination	2	7
4	14	The nightspire	4	9
5	15	The Frozen Throne	3	8
6	16	Court of Stars	5	10



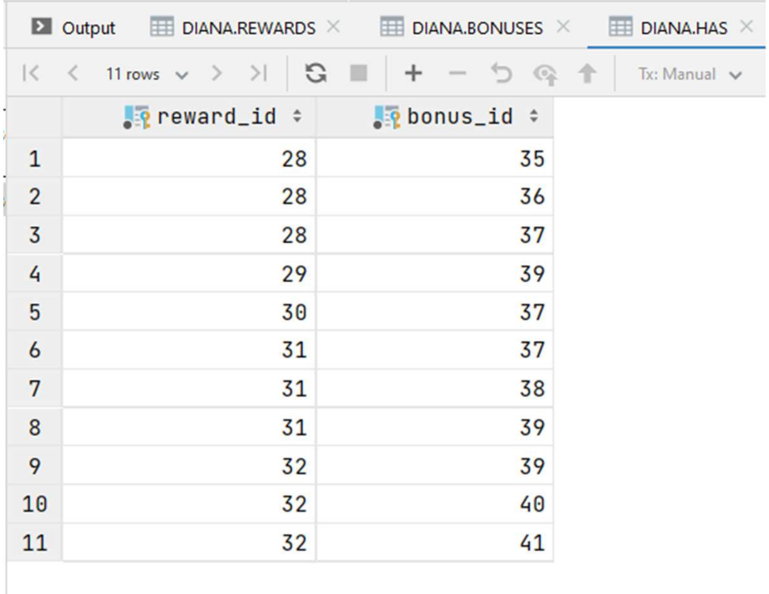
	reward_id	"name"	description	sell_price	boss_id
1	28	Bindings of the Herald	Binds when picked up	320	7
2	29	The Lady's Necklace	The necklace shimmers, a remarkable piece of jewelry that doesn…	1000	7
3	30	Convergence of Fates	Binds when picked up	400	9
4	31	Mana-Tinged Pack	Contains assorted potions and magical reagents from the Broken …	50	10
5	32	Warmace of Menethil	Binds when picked up	430	8
6	33	Cursed Lover's Ring	No not touch! Ups I've touched it	300	6
7	34	Banshee's Bow	A bow made out of a human spine. It has o dark violet glow.	<null>	7



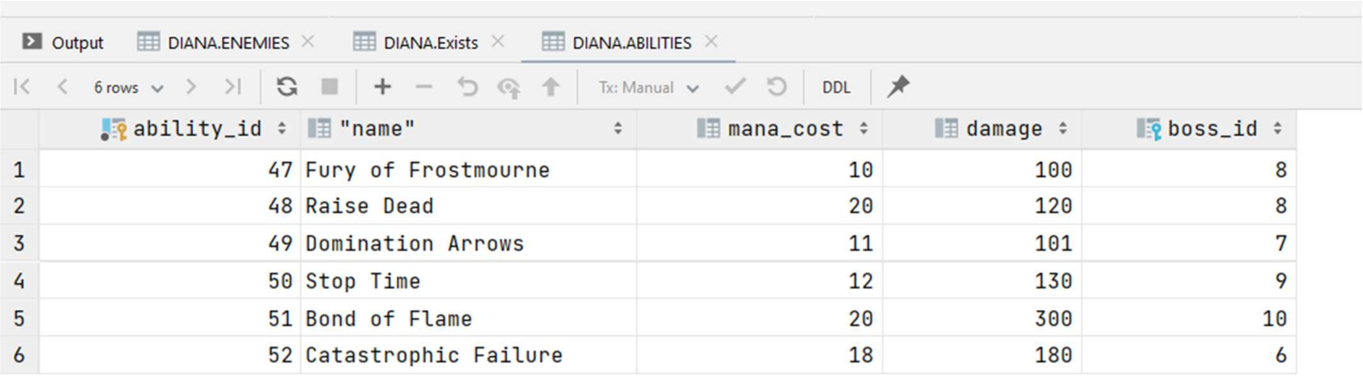
	boss_id	"name"	health	damage	mana	dificulty
1	6	Queen Azhara	121000	4000	100	<null>
2	7	Sylvanas Windrunner	143000	6000	400	<null>
3	8	Lich King	4300	300	40	Medium
4	9	Elisande	6300	700	60	Easy
5	10	Talixae Flamewreath	100000	3000	600	<null>



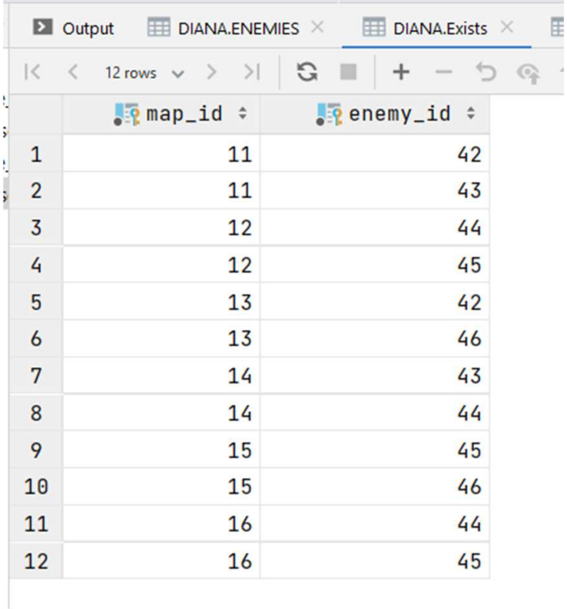
	bonus_id	"name"	quantity
1	35	Stamina	18
2	36	Haste	39
3	37	Versatility	29
4	38	Magic	100
5	39	Stamina	50
6	40	Atack speed	20
7	41	Mastery	30



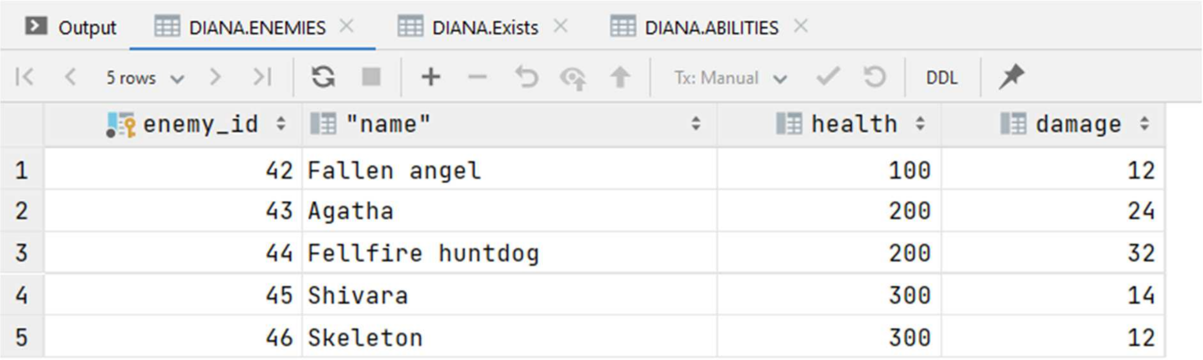
	reward_id	bonus_id
1	28	35
2	28	36
3	28	37
4	29	39
5	30	37
6	31	37
7	31	38
8	31	39
9	32	39
10	32	40
11	32	41



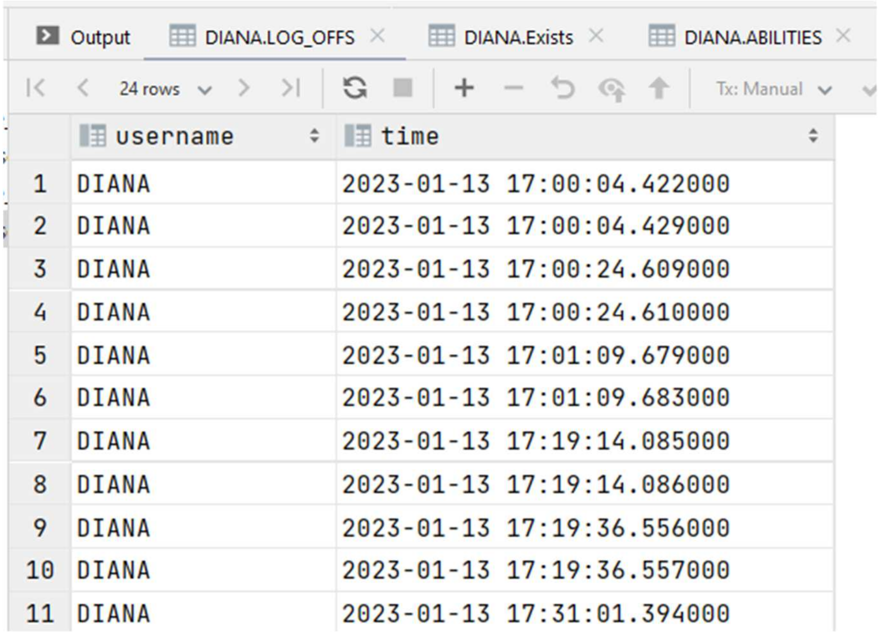
	ability_id	"name"	mana_cost	damage	boss_id
1	47	Fury of Frostmourne	10	100	8
2	48	Raise Dead	20	120	8
3	49	Domination Arrows	11	101	7
4	50	Stop Time	12	130	9
5	51	Bond of Flame	20	300	10
6	52	Catastrophic Failure	18	180	6



	map_id	enemy_id
1	11	42
2	11	43
3	12	44
4	12	45
5	13	42
6	13	46
7	14	43
8	14	44
9	15	45
10	15	46
11	16	44
12	16	45



	enemy_id	"name"	health	damage
1	42	Fallen angel	100	12
2	43	Agatha	200	24
3	44	Fellfire huntdog	200	32
4	45	Shivara	300	14
5	46	Skeleton	300	12



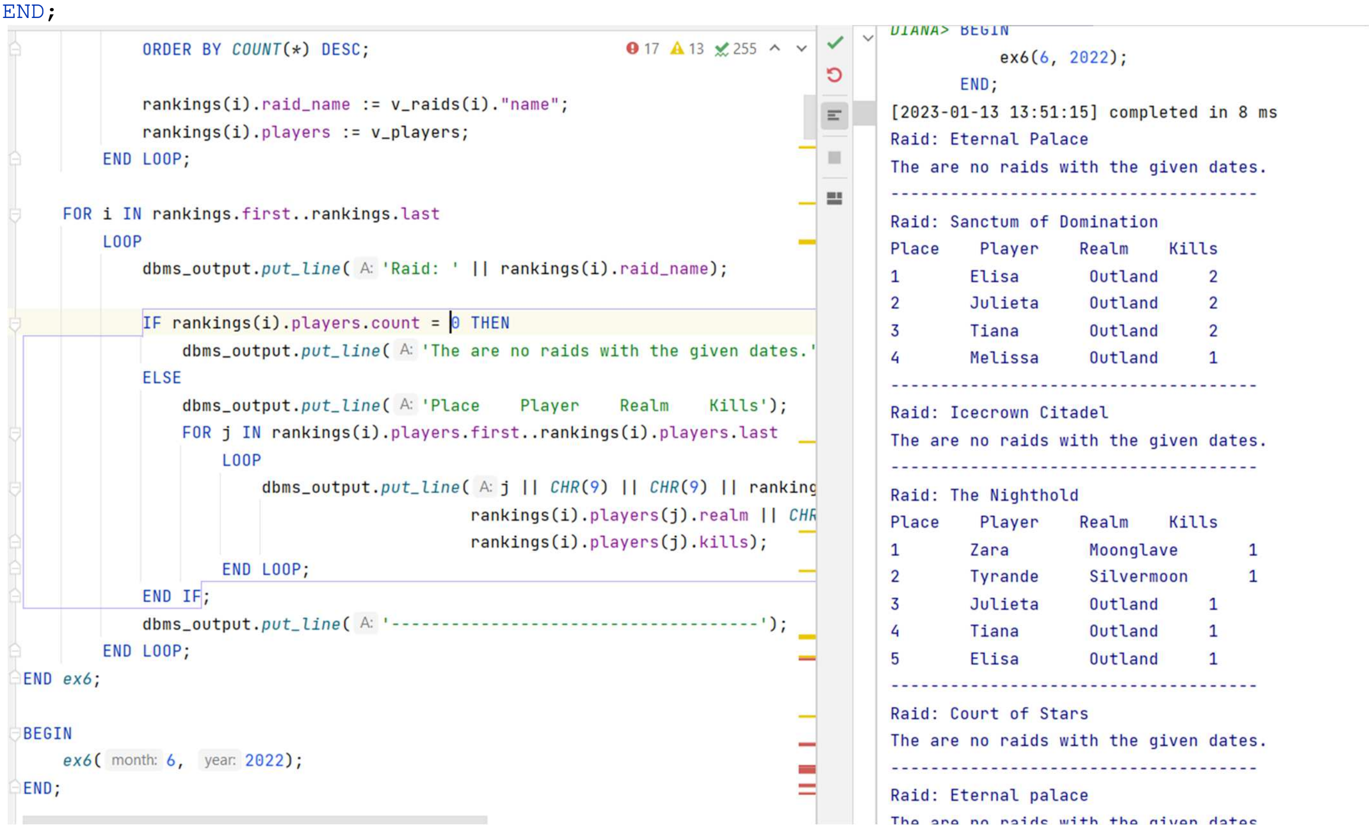
	username	time
1	DIANA	2023-01-13 17:00:04.422000
2	DIANA	2023-01-13 17:00:04.429000
3	DIANA	2023-01-13 17:00:24.609000
4	DIANA	2023-01-13 17:00:24.610000
5	DIANA	2023-01-13 17:01:09.679000
6	DIANA	2023-01-13 17:01:09.683000
7	DIANA	2023-01-13 17:19:14.085000
8	DIANA	2023-01-13 17:19:14.086000
9	DIANA	2023-01-13 17:19:36.556000
10	DIANA	2023-01-13 17:19:36.557000
11	DIANA	2023-01-13 17:31:01.394000

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.

Pentru fiecare raid să se afișeze clasamentul jucătorilor din luna și anul specificat. Clasamentul se face după numărul de boss-i înfrânți și mai mulți jucători nu pot împărți același loc. Pentru un jucător se vor afișa numele, realm-ul și numărul de boss-i înfrânți.

- Coletii utilizare: **tabel indexat de înregistrări care conțin un vector**, **vector**, **înregistrare**
- Subprogram stocat de tip **procedură**.

```
CREATE OR REPLACE PROCEDURE ex6(month NUMBER, year NUMBER)
    IS
    TYPE player_rec IS RECORD
                       (
                           name  players."name"%TYPE,
                           realm players.realm%TYPE,
                           kills NUMBER
                       );
    TYPE vect_players IS VARRAY(1000) OF player_rec;
    TYPE raid_ranking IS RECORD
                         (
                             raid_name raids."name"%TYPE,
                             players   vect_players
                         );
    -- colectie 1 : colectie imbricata care contine numele raidului si lista de jucatori a
acelui raid
    TYPE rankings_tbl IS TABLE OF raid_ranking
        INDEX BY BINARY_INTEGER;
    rankings  rankings_tbl;
    v_players vect_players;
    -- colectie 2: vector de raiduri
    TYPE raids_vect IS VARRAY(100) OF raids%ROWTYPE;
    v_raids   raids_vect;
BEGIN
    SELECT * BULK COLLECT
    INTO v_raids
    FROM raids;

    FOR i IN v_raids.first..v_raids.last
        LOOP
            SELECT MAX(players."name"),
                   MAX(players.realm),
                   COUNT(*) BULK COLLECT
            INTO v_players
            FROM players
                     JOIN participates USING (player_id)
                     JOIN instances USING (instance_id)
                     JOIN raids USING (raid_id)
                     JOIN maps USING (raid_id)
                     JOIN bosses USING (boss_id)
            WHERE raid_id = v_raids(i).raid_id
              AND EXTRACT(MONTH FROM instances.create_date) = month
              AND EXTRACT(YEAR FROM instances.create_date) = year
            GROUP BY player_id
            ORDER BY COUNT(*) DESC;

            rankings(i).raid_name := v_raids(i)."name";
            rankings(i).players := v_players;
        END LOOP;

    FOR i IN rankings.first..rankings.last
        LOOP
            dbms_output.put_line('Raid: ' || rankings(i).raid_name);

            IF rankings(i).players.count = 0 THEN
                dbms_output.put_line('The are no raids with the given dates.');
            ELSE
                dbms_output.put_line('Place    Player    Realm    Kills');
                FOR j IN rankings(i).players.first..rankings(i).players.last
                    LOOP
                        dbms_output.put_line(j || CHR(9) || CHR(9) ||
rankings(i).players(j).name || CHR(9) || CHR(9) ||
                                             rankings(i).players(j).realm || CHR(9) || CHR(9)
||
                                                 rankings(i).players(j).kills);
                         END LOOP;
                END IF;
                dbms_output.put_line('-------------------------------------');
            END LOOP;
END ex6;

BEGIN
    ex6(6, 2022);
END;
```



```
            ORDER BY COUNT(*) DESC;

            rankings(i).raid_name := v_raids(i)."name";
            rankings(i).players := v_players;
        END LOOP;

    FOR i IN rankings.first..rankings.last
        LOOP
            dbms_output.put_line( A: 'Raid: ' || rankings(i).raid_name);

            IF rankings(i).players.count = 0 THEN
                dbms_output.put_line( A: 'The are no raids with the given dates.'
            ELSE
                dbms_output.put_line( A: 'Place     Player     Realm     Kills');
                FOR j IN rankings(i).players.first..rankings(i).players.last
                    LOOP
                        dbms_output.put_line( A: j || CHR(9) || CHR(9) || ranking
                                                     rankings(i).players(j).realm || CHR
                                                     rankings(i).players(j).kills);
                    END LOOP;
            END IF;
            dbms_output.put_line( A: '-------------------------------------');
        END LOOP;
END ex6;

BEGIN
    ex6( month: 6,  year: 2022);
END;
```

```
DIANA> BEGIN
            ex6(6, 2022);
        END;
[2023-01-13 13:51:15] completed in 8 ms
Raid: Eternal Palace
The are no raids with the given dates.
-------------------------------------
Raid: Sanctum of Domination
Place    Player    Realm    Kills
1       Elisa      Outland    2
2       Julieta    Outland    2
3       Tiana      Outland    2
4       Melissa    Outland    1
-------------------------------------
Raid: Icecrown Citadel
The are no raids with the given dates.
-------------------------------------
Raid: The Nighthold
Place    Player    Realm    Kills
1       Zara       Moonglave       1
2       Tyrande    Silvermoon      1
3       Julieta    Outland    1
4       Tiana      Outland    1
5       Elisa      Outland    1
-------------------------------------
Raid: Court of Stars
The are no raids with the given dates.
-------------------------------------
Raid: Eternal palace
The are no raids with the given dates
```

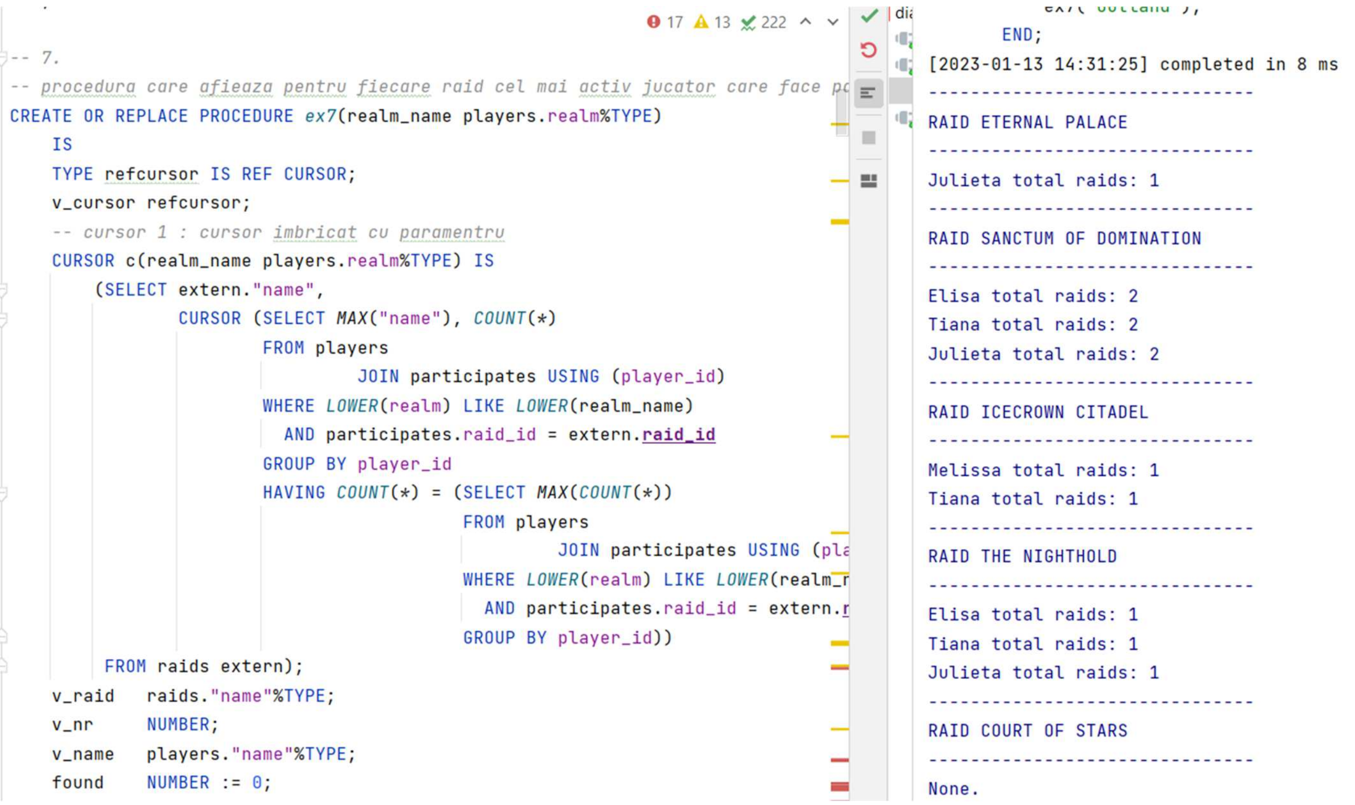
7. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un **subprogram stocat independent** care să utilizeze 2 tipuri diferite de cursoare studiate, unul dintre acestea fiind cursor parametrizat. Apelați subprogramul.

Pentru fiecare raid să se afișeze cel mai activ jucător care face parte din reaml-ul dat ca parametru.

- Subprogram stocat de tip **procedură**
- Tipuri de cursoare: **expresie cursor cu paramentru**, **ciclu cursor**

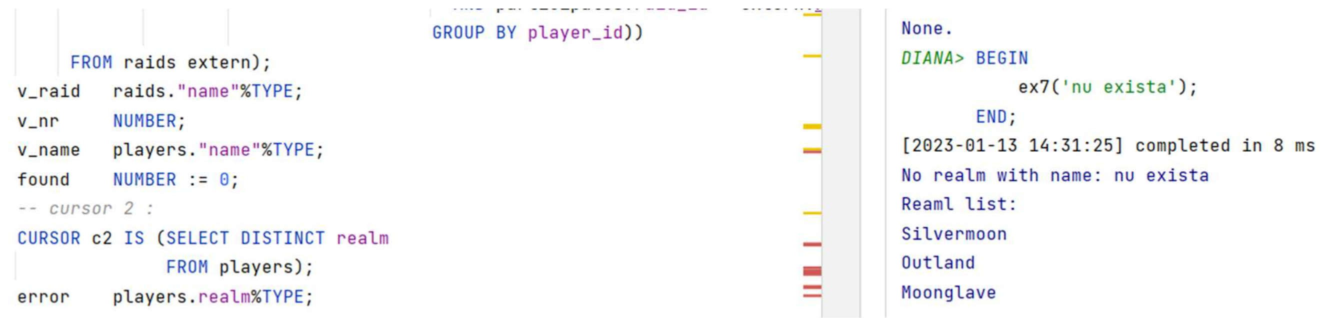
```
CREATE OR REPLACE PROCEDURE ex7(realm_name players.realm%TYPE)
  IS
  TYPE refcursor IS REF CURSOR;
  v_cursor refcursor;
  -- cursor 1 : cursor imbricat cu paramentru
  CURSOR c(realm_name players.realm%TYPE) IS
       (SELECT extern."name",
               CURSOR (SELECT MAX("name"), COUNT(*)
                       FROM players
                                JOIN participates USING (player_id)
                       WHERE LOWER(realm) LIKE LOWER(realm_name)
                         AND participates.raid_id = extern.raid_id
                       GROUP BY player_id
                       HAVING COUNT(*) = (SELECT MAX(COUNT(*))
                                          FROM players
                                                   JOIN participates USING (player_id)
                                          WHERE LOWER(realm) LIKE LOWER(realm_name)
                                            AND participates.raid_id = extern.raid_id
                                          GROUP BY player_id))
        FROM raids extern);
  v_raid   raids."name"%TYPE;
  v_nr     NUMBER;
  v_name   players."name"%TYPE;
```

```
    found    NUMBER := 0;
    -- cursor 2 :
    CURSOR c2 IS (SELECT DISTINCT realm
                  FROM players);
    error    players.realm%TYPE;
    no_realm EXCEPTION;
    PRAGMA EXCEPTION_INIT (no_realm, -9090);
BEGIN
    SELECT COUNT(*)
    INTO error
    FROM players
    WHERE LOWER(realm) LIKE LOWER(realm_name);

    IF error = 0 THEN
        RAISE no_realm;
    END IF;

    OPEN c(realm_name);
    LOOP
        FETCH c INTO v_raid, v_cursor;
        EXIT WHEN c%NOTFOUND;
        dbms_output.put_line('--------------------------------');
        dbms_output.put_line('RAID ' || UPPER(v_raid));
        dbms_output.put_line('--------------------------------');
        found := 0;
        LOOP
            FETCH v_cursor INTO v_name, v_nr;
            EXIT WHEN v_cursor%NOTFOUND;
            found := 1;
            dbms_output.put_line(v_name || ' total raids: ' || v_nr);
        END LOOP;
        IF found = 0 THEN
            dbms_output.put_line('None.');
        END IF;
    END LOOP;
    CLOSE c;
EXCEPTION
    WHEN no_realm THEN
        dbms_output.put_line('No realm with name: ' || realm_name);
        dbms_output.put_line('Reaml list: ');
        for r in c2 LOOP
            dbms_output.put_line(r.realm);
            END LOOP;
END ex7;


BEGIN
    ex7('Outland');
END;

BEGIN
    ex7('nu exista');
END;
```



```
-- 7.
-- procedura care afieaza pentru fiecare raid cel mai activ jucator care face pc
CREATE OR REPLACE PROCEDURE ex7(realm_name players.realm%TYPE)
    IS
    TYPE refcursor IS REF CURSOR;
    v_cursor refcursor;
    -- cursor 1 : cursor imbricat cu paramentru
    CURSOR c(realm_name players.realm%TYPE) IS
        (SELECT extern."name",
                CURSOR (SELECT MAX("name"), COUNT(*)
                        FROM players
                                JOIN participates USING (player_id)
                        WHERE LOWER(realm) LIKE LOWER(realm_name)
                          AND participates.raid_id = extern.raid_id
                        GROUP BY player_id
                        HAVING COUNT(*) = (SELECT MAX(COUNT(*))
                                           FROM players
                                                    JOIN participates USING (pla
                                           WHERE LOWER(realm) LIKE LOWER(realm_r
                                             AND participates.raid_id = extern.r
                                           GROUP BY player_id))
         FROM raids extern);
    v_raid   raids."name"%TYPE;
    v_nr     NUMBER;
    v_name   players."name"%TYPE;
    found    NUMBER := 0;
```

```
            ex7('Outland');
        END;
[2023-01-13 14:31:25] completed in 8 ms
------------------------------
RAID ETERNAL PALACE
------------------------------
Julieta total raids: 1
------------------------------
RAID SANCTUM OF DOMINATION
------------------------------
Elisa total raids: 2
Tiana total raids: 2
Julieta total raids: 2
------------------------------
RAID ICECROWN CITADEL
------------------------------
Melissa total raids: 1
Tiana total raids: 1
------------------------------
RAID THE NIGHTHOLD
------------------------------
Elisa total raids: 1
Tiana total raids: 1
Julieta total raids: 1
------------------------------
RAID COURT OF STARS
------------------------------
None.
```



```
                                           GROUP BY player_id))
         FROM raids extern);
    v_raid   raids."name"%TYPE;
    v_nr     NUMBER;
    v_name   players."name"%TYPE;
    found    NUMBER := 0;
    -- cursor 2 :
    CURSOR c2 IS (SELECT DISTINCT realm
                  FROM players);
    error    players.realm%TYPE;
```

```
None.
DIANA> BEGIN
            ex7('nu exista');
        END;
[2023-01-13 14:31:25] completed in 8 ms
No realm with name: nu exista
Reaml list:
Silvermoon
Outland
Moonglave
```

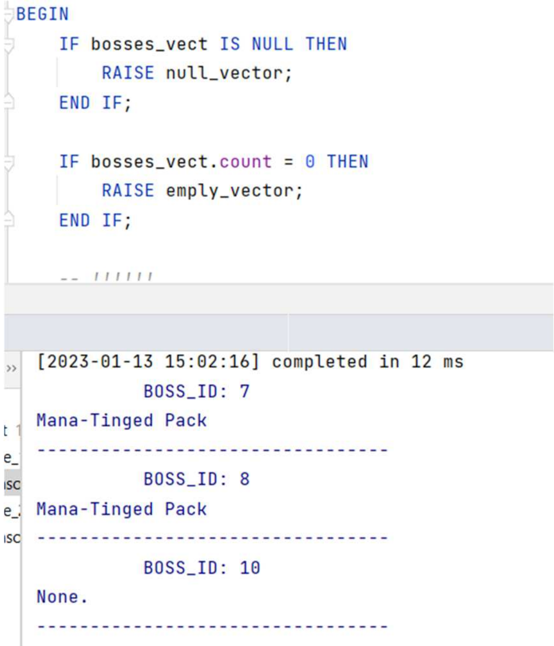
8. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un **subprogram stocat independent de tip funcție** care să utilizeze **într-o singură comandă SQL** 3 dintre tabelele definite. Definiți minim 2 excepții. Apelați subprogramul astfel încât să evidențiați **toate** cazurile tratate.

Se dă un vector de id-uri. Pentru fiecare boss din vectorul de id-uri să se afle din mulțimea totală a recompenselor recompensele care au măcar un bonus mai mare decât toate bonusurile oferite de boss-ul curent.

- Tabele ultilizate: **rewards**, **has**, **bonuses**
- Excepții: **boss_not_found**, **emply_vector**, **null_vector**
- Subprogram stocat de tip **funcție** care returnează un tabel imbricat de vectori

```
CREATE OR REPLACE TYPE vect_b IS VARRAY(5000) OF NUMBER(38);
CREATE OR REPLACE TYPE vect_r IS TABLE OF VARCHAR(255);
CREATE OR REPLACE TYPE tabel IS TABLE OF vect_r;
/

CREATE OR REPLACE FUNCTION ex8(bosses_vect vect_b)
    RETURN tabel IS

    -- 3 exceptii
    boss_not_found EXCEPTION;
    PRAGMA EXCEPTION_INIT (boss_not_found, -2000);

    emply_vector EXCEPTION;
    PRAGMA EXCEPTION_INIT (emply_vector, -3000);

    null_vector EXCEPTION;
    PRAGMA EXCEPTION_INIT (null_vector, -4000);
    nr      NUMBER;
    maxi   bosses.boss_id%TYPE;
    vector vect_r := vect_r();
    output tabel  := tabel();
BEGIN
    IF bosses_vect IS NULL THEN
        RAISE null_vector;
    END IF;

    IF bosses_vect.count = 0 THEN
        RAISE emply_vector;
    END IF;

    -- !!!!!!
    SELECT MAX(boss_id)
    INTO maxi
    FROM bosses;

    FOR i IN 1..maxi
        LOOP
            output.extend;
        END LOOP;

    FOR i IN bosses_vect.first..bosses_vect.last
        LOOP
            SELECT COUNT(*)
            INTO nr
            FROM bosses
            WHERE boss_id = bosses_vect(i);

            IF nr = 0 THEN
                RAISE boss_not_found;
            END IF;

            -- comanda cu 3 tabele
            SELECT DISTINCT rewards."name" BULK COLLECT
            INTO vector
            FROM rewards
                     JOIN has ON has.reward_id = rewards.reward_id
                     JOIN bonuses ON bonuses.bonus_id = has.bonus_id
            WHERE bonuses.quantity > ALL (SELECT bonuses.quantity
                                          FROM rewards
                                                   JOIN has ON has.reward_id =
rewards.reward_id
                                                   JOIN bonuses ON bonuses.bonus_id =
has.bonus_id
                                          WHERE rewards.boss_id = bosses_vect(i));
            IF vector IS NULL THEN
                output(bosses_vect(i)) := vect_r();
            ELSE
                output(bosses_vect(i)) := vector;
            END IF;
        END LOOP;
    RETURN output;
EXCEPTION
    WHEN emply_vector THEN
        dbms_output.put_line('Vector is empty.');
        RETURN NULL;
    WHEN null_vector THEN
        dbms_output.put_line('Vector is null.');
        RETURN NULL;
    WHEN boss_not_found THEN
        dbms_output.put_line('No boss found.');
        RETURN NULL;
END ex8;
-- corect
DECLARE
    rezultat tabel  := tabel();
    -- corect
    input1   vect_b := vect_b(7, 8, 10);
BEGIN
    rezultat := ex8(input1);
    IF rezultat IS NOT NULL THEN
        FOR i IN rezultat.first..rezultat.last
            LOOP
                IF rezultat(i) IS NOT NULL THEN
```

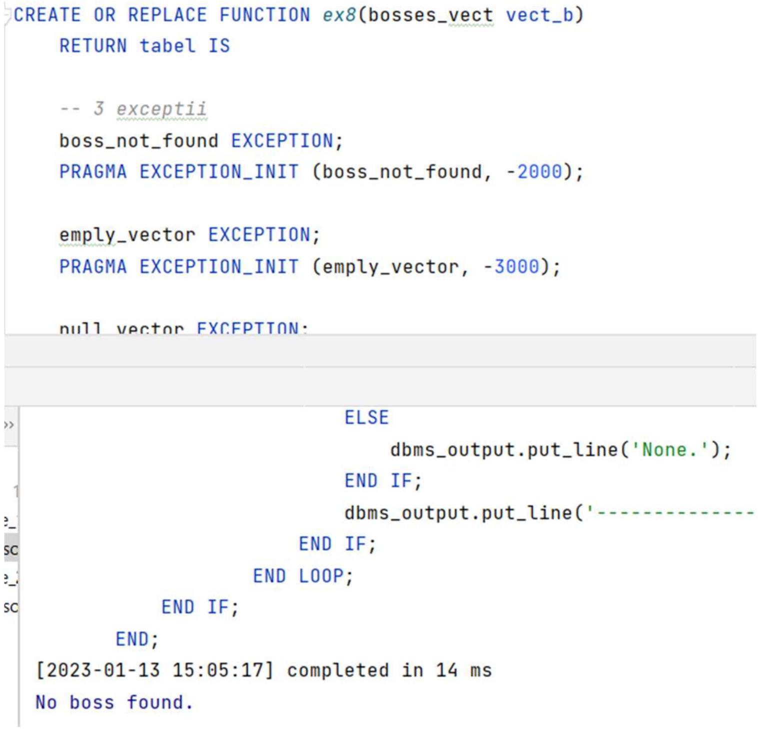
```
                    dbms_output.put_line('          BOSS_ID: ' || i || '          ');

                    IF rezultat(i).count != 0 THEN
                        FOR j IN rezultat(i).first..rezultat(i).last
                            LOOP
                                dbms_output.put_line(rezultat(i)(j));
                            END LOOP;
                    ELSE
                        dbms_output.put_line('None.');
                    END IF;
                    dbms_output.put_line('---------------------------------');
                END IF;
            END LOOP;
    END IF;
END;
```



```
BEGIN
    IF bosses_vect IS NULL THEN
        RAISE null_vector;
    END IF;

    IF bosses_vect.count = 0 THEN
        RAISE emply_vector;
    END IF;

    -- ''''''
```

```
[2023-01-13 15:02:16] completed in 12 ms
          BOSS_ID: 7
Mana-Tinged Pack
---------------------------------
          BOSS_ID: 8
Mana-Tinged Pack
---------------------------------
          BOSS_ID: 10
None.
---------------------------------
```

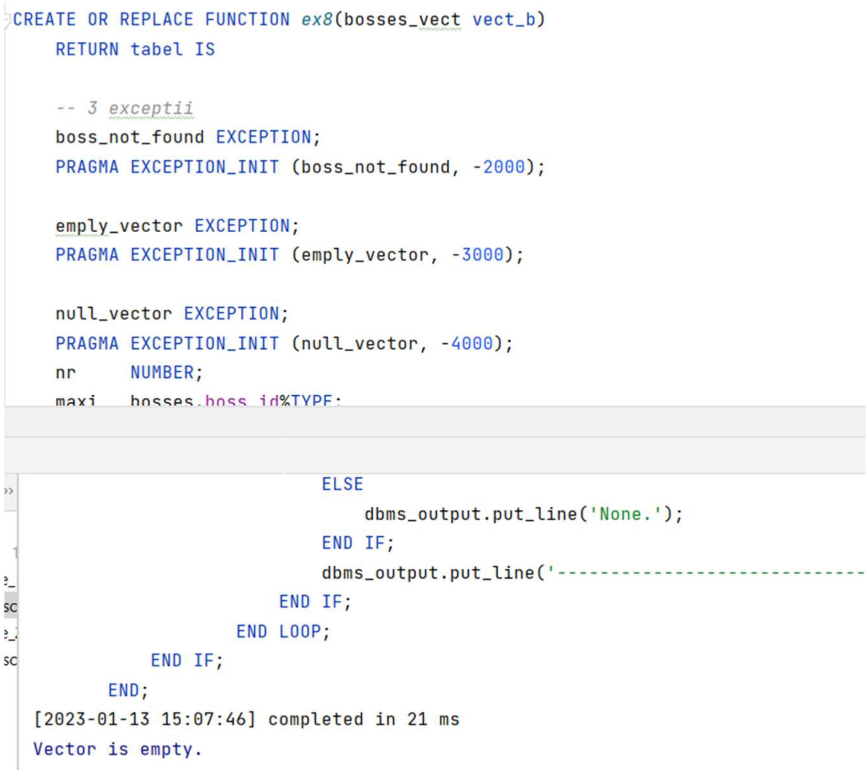
```
-- boss not found
DECLARE
    rezultat tabel  := tabel();
    -- boss not found
    input2   vect_b := vect_b(7, 8, 100);
BEGIN
    rezultat := ex8(input2);
    IF rezultat IS NOT NULL THEN
        FOR i IN rezultat.first..rezultat.last
            LOOP
                IF rezultat(i) IS NOT NULL THEN
                    dbms_output.put_line('          BOSS_ID: ' || i || '          ');

                    IF rezultat(i).count != 0 THEN
                        FOR j IN rezultat(i).first..rezultat(i).last
                            LOOP
                                dbms_output.put_line(rezultat(i)(j));
                            END LOOP;
                    ELSE
                        dbms_output.put_line('None.');
                    END IF;
                    dbms_output.put_line('---------------------------------');
                END IF;
            END LOOP;
    END IF;
END;
```



```
CREATE OR REPLACE FUNCTION ex8(bosses_vect vect_b)
    RETURN tabel IS

    -- 3 exceptii
    boss_not_found EXCEPTION;
    PRAGMA EXCEPTION_INIT (boss_not_found, -2000);

    emply_vector EXCEPTION;
    PRAGMA EXCEPTION_INIT (emply_vector, -3000);

    null_vector EXCEPTION;
```

```
                         ELSE
                             dbms_output.put_line('None.');
                         END IF;
                         dbms_output.put_line('--------------
                     END IF;
                 END LOOP;
             END IF;
         END;
[2023-01-13 15:05:17] completed in 14 ms
No boss found.
```

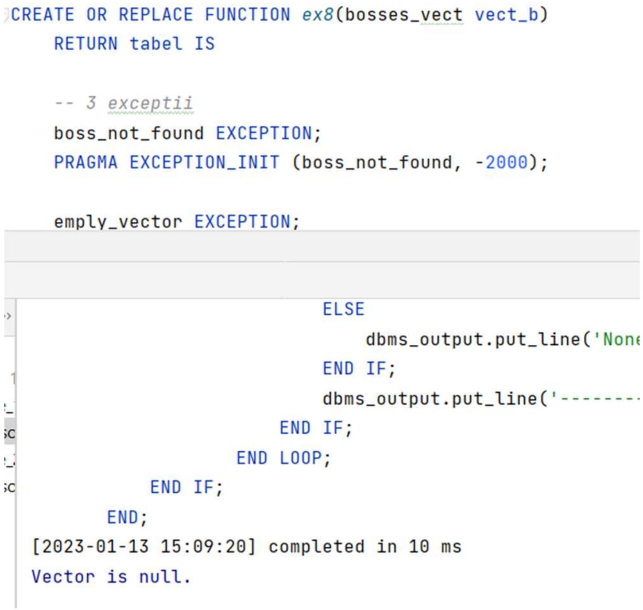
```
-- empty vector
DECLARE
    rezultat tabel  := tabel();
    -- empty vector
    input3   vect_b := vect_b();
BEGIN
    rezultat := ex8(input3);
    IF rezultat IS NOT NULL THEN
        FOR i IN rezultat.first..rezultat.last
            LOOP
                IF rezultat(i) IS NOT NULL THEN
                    dbms_output.put_line('          BOSS_ID: ' || i || '           ');

                    IF rezultat(i).count != 0 THEN
                        FOR j IN rezultat(i).first..rezultat(i).last
                            LOOP
                                dbms_output.put_line(rezultat(i)(j));
                            END LOOP;
                    ELSE
                        dbms_output.put_line('None.');
                    END IF;
                    dbms_output.put_line('---------------------------------');
                END IF;
            END LOOP;
    END IF;
END;
```



```
CREATE OR REPLACE FUNCTION ex8(bosses_vect vect_b)
    RETURN tabel IS

    -- 3 exceptii
    boss_not_found EXCEPTION;
    PRAGMA EXCEPTION_INIT (boss_not_found, -2000);

    emply_vector EXCEPTION;
    PRAGMA EXCEPTION_INIT (emply_vector, -3000);

    null_vector EXCEPTION;
    PRAGMA EXCEPTION_INIT (null_vector, -4000);
    nr     NUMBER;
    maxi   bosses.boss_id%TYPE;
```

```
                               ELSE
                                   dbms_output.put_line('None.');
                               END IF;
                               dbms_output.put_line('-----------------------------
                           END IF;
                       END LOOP;
               END IF;
           END;
[2023-01-13 15:07:46] completed in 21 ms
Vector is empty.
```

```
-- null vector
DECLARE
    rezultat tabel  := tabel();
    -- null vector
    input4   vect_b;
BEGIN
    rezultat := ex8(input4);
    IF rezultat IS NOT NULL THEN
        FOR i IN rezultat.first..rezultat.last
            LOOP
                IF rezultat(i) IS NOT NULL THEN
                    dbms_output.put_line('          BOSS_ID: ' || i || '          ');

                    IF rezultat(i).count != 0 THEN
                        FOR j IN rezultat(i).first..rezultat(i).last
                            LOOP
                                dbms_output.put_line(rezultat(i)(j));
                            END LOOP;
                    ELSE
                        dbms_output.put_line('None.');
                    END IF;
                    dbms_output.put_line('---------------------------------');
                END IF;
            END LOOP;
    END IF;
END;
```



```
CREATE OR REPLACE FUNCTION ex8(bosses_vect vect_b)
    RETURN tabel IS

    -- 3 exceptii
    boss_not_found EXCEPTION;
    PRAGMA EXCEPTION_INIT (boss_not_found, -2000);

    emply_vector EXCEPTION;
```

```
                               ELSE
                                   dbms_output.put_line('None
                               END IF;
                               dbms_output.put_line('-------
                           END IF;
                       END LOOP;
               END IF;
           END;
[2023-01-13 15:09:20] completed in 10 ms
Vector is null.
```

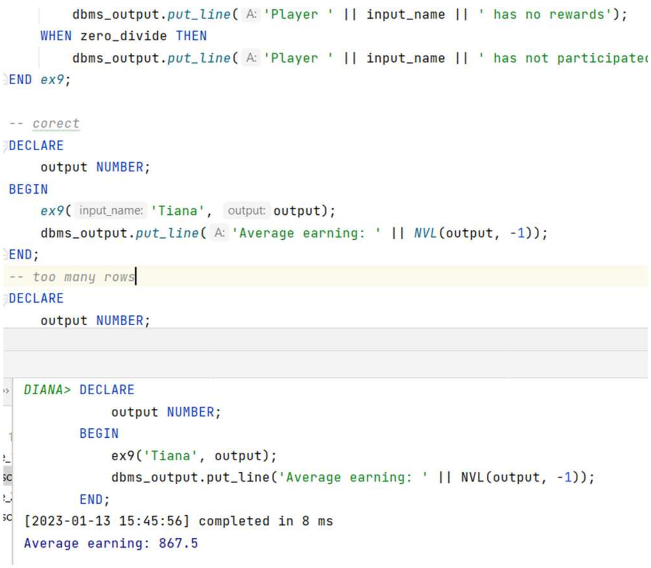
9. Formulați în limbaj natural o problemă pe care să o rezolvați folosind un **subprogram stocat independent de tip procedură** care să utilizeze **într-o singură comandă SQL** 5 dintre tabelele definite. Tratați toate excepțiile care pot apărea, incluzând excepțiile NO_DATA_FOUND și TOO_MANY_ROWS. Apelați subprogramul astfel încât să evidențiați toate cazurile tratate.

Pentru un jucător să se afle câștigul mediu (câștig mediu = câștig / nr_raiduri_la_care_a_participat).

- Subprogram stocat de tip **procedură** care returnează rezultatul printr-un paramentru OUT
- Tabele folosite: **participates**, **raids**, **bosses**, **rewards**, **maps**
- Exceții: NO_DATA_FOUND, TOO_MANY_ROWS, **no_rewards**, **zero_divide**

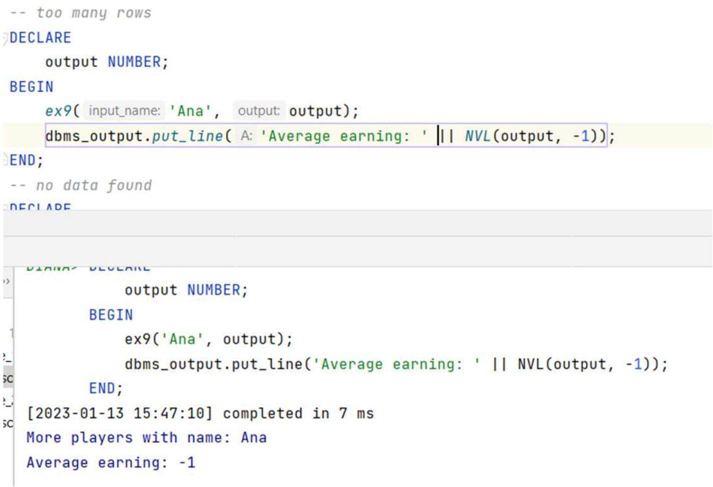
```
-- pentru a testa mai introduc niste date.
INSERT INTO players
VALUES (10000, 'Maia', 'Outland', 'Night elf', 'F', 'mage', 'damage-dealer', 60);
INSERT INTO players
VALUES (10001, 'Ana', 'Outland', 'Night elf', 'F', 'mage', 'damage-dealer', 60);
INSERT INTO players
VALUES (10002, 'Ana', 'Outland', 'Night elf', 'F', 'mage', 'damage-dealer', 60);
INSERT INTO players
VALUES (10003, 'Camila', 'Outland', 'Night elf', 'F', 'mage', 'damage-dealer', 60);

INSERT INTO instances
VALUES (100, SYSDATE);

INSERT INTO raids
VALUES (1001,'raid','locatie',30,40);

INSERT INTO participates
VALUES (1001, 10000, 100);


CREATE OR REPLACE PROCEDURE ex9(input_name players."name"%TYPE, output OUT NUMBER)
    IS
    id        players.player_id%TYPE;
    raids_nr NUMBER;
    v_sum    NUMBER;
    no_rewards EXCEPTION;
    PRAGMA EXCEPTION_INIT (no_rewards, -3000);
BEGIN
    SELECT player_id
    INTO id
    FROM players
    WHERE "name" = input_name;

    SELECT COUNT(*)
    INTO raids_nr
    FROM participates
    WHERE player_id = id;

    SELECT SUM(NVL(sell_price, 0))
    INTO v_sum
    FROM participates
             JOIN raids USING (raid_id)
             JOIN maps USING (raid_id)
             JOIN bosses USING (boss_id)
             JOIN rewards USING (boss_id)
    WHERE player_id = id;

    output := NVL(v_sum, 0) / raids_nr;

    IF v_sum IS NULL THEN
        RAISE no_rewards;
    END IF;

EXCEPTION
    WHEN too_many_rows THEN
        dbms_output.put_line('More players with name: ' || input_name);
    WHEN no_data_found THEN
        dbms_output.put_line('No player with name: ' || input_name);
    WHEN no_rewards THEN
        dbms_output.put_line('Player ' || input_name || ' has no rewards');
    WHEN zero_divide THEN
        dbms_output.put_line('Player ' || input_name || ' has not participated to any
raid yet');
END ex9;
```

```
-- corect
DECLARE
    output NUMBER;
BEGIN
    ex9('Tiana', output);
    dbms_output.put_line('Average earning: ' || NVL(output, -1));
END;
-- too many rows
DECLARE
    output NUMBER;
BEGIN
    ex9('Ana', output);
    dbms_output.put_line('Average earning: ' || NVL(output, -1));
END;
-- no data found
DECLARE
    output NUMBER;
BEGIN
    ex9('nu exista', output);
    dbms_output.put_line('Average earning: ' || NVL(output, -1));
END;
-- no rewards
DECLARE
    output NUMBER;
BEGIN
    ex9('Maia', output);
    dbms_output.put_line('Average earning: ' || NVL(output, -1));
END;
-- zero divide
DECLARE
    output NUMBER;
BEGIN
    ex9('Camila', output);
    dbms_output.put_line('Average earning: ' || NVL(output, -1));
END;
```



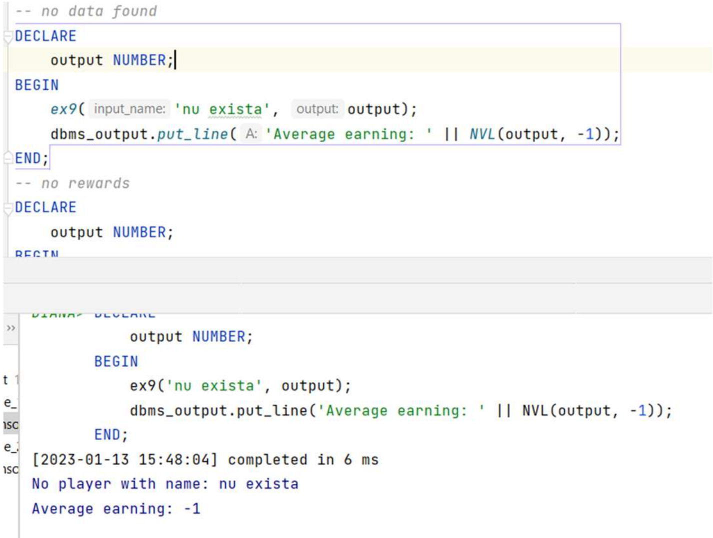
```
        dbms_output.put_line( A: 'Player ' || input_name || ' has no rewards');
    WHEN zero_divide THEN
        dbms_output.put_line( A: 'Player ' || input_name || ' has not participatec
END ex9;

-- corect
DECLARE
    output NUMBER;
BEGIN
    ex9( input_name: 'Tiana',  output: output);
    dbms_output.put_line( A: 'Average earning: ' || NVL(output, -1));
END;
-- too many rows
DECLARE
    output NUMBER;
```

```
DIANA> DECLARE
           output NUMBER;
       BEGIN
           ex9('Tiana', output);
           dbms_output.put_line('Average earning: ' || NVL(output, -1));
       END;
[2023-01-13 15:45:56] completed in 8 ms
Average earning: 867.5
```



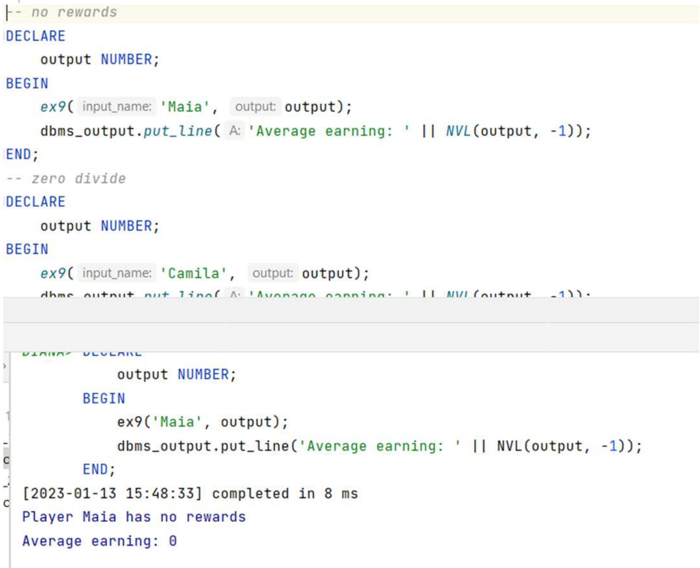
```
-- too many rows
DECLARE
    output NUMBER;
BEGIN
    ex9( input_name: 'Ana',  output: output);
    dbms_output.put_line( A: 'Average earning: ' || NVL(output, -1));
END;
-- no data found
DECLARE
```

```
DIANA> DECLARE
           output NUMBER;
       BEGIN
           ex9('Ana', output);
           dbms_output.put_line('Average earning: ' || NVL(output, -1));
       END;
[2023-01-13 15:47:10] completed in 7 ms
More players with name: Ana
Average earning: -1
```



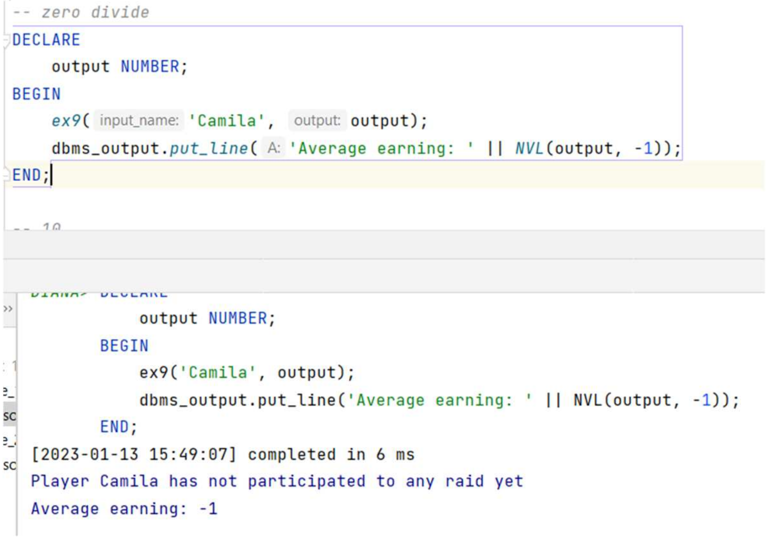
```
-- no data found
DECLARE
    output NUMBER;
BEGIN
    ex9( input_name: 'nu exista',  output: output);
    dbms_output.put_line( A: 'Average earning: ' || NVL(output, -1));
END;
-- no rewards
DECLARE
    output NUMBER;
BEGIN
```

```
DIANA> DECLARE
           output NUMBER;
       BEGIN
           ex9('nu exista', output);
           dbms_output.put_line('Average earning: ' || NVL(output, -1));
       END;
[2023-01-13 15:48:04] completed in 6 ms
No player with name: nu exista
Average earning: -1
```



```
-- no rewards
DECLARE
    output NUMBER;
BEGIN
    ex9( input_name: 'Maia',  output: output);
    dbms_output.put_line( A: 'Average earning: ' || NVL(output, -1));
END;
-- zero divide
DECLARE
    output NUMBER;
BEGIN
    ex9( input_name: 'Camila',  output: output);
    dbms_output.put_line( A: 'Average earning: ' || NVL(output, -1));
```

```
DIANA> DECLARE
           output NUMBER;
       BEGIN
           ex9('Maia', output);
           dbms_output.put_line('Average earning: ' || NVL(output, -1));
       END;
[2023-01-13 15:48:33] completed in 8 ms
Player Maia has no rewards
Average earning: 0
```



```
-- zero divide
DECLARE
    output NUMBER;
BEGIN
    ex9( input_name: 'Camila',  output: output);
    dbms_output.put_line( A: 'Average earning: ' || NVL(output, -1));
END;

-- 10
```

```
DIANA> DECLARE
           output NUMBER;
       BEGIN
           ex9('Camila', output);
           dbms_output.put_line('Average earning: ' || NVL(output, -1));
       END;
[2023-01-13 15:49:07] completed in 6 ms
Player Camila has not participated to any raid yet
Average earning: -1
```

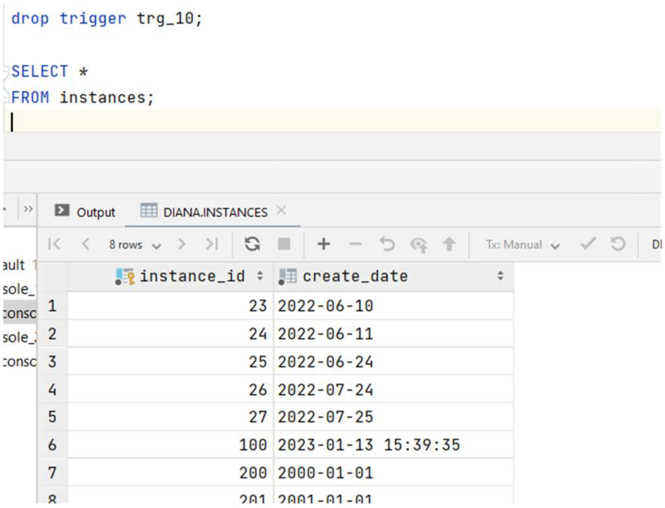
10. Definiți un *trigger* de tip LMD la nivel de comandă. Declanșați *trigger*-ul.

Definesc un trigger care de fiecare dată cănd se adaugă o instanță nouă le sterge pe cele expirate (care sunt mai vechi de 5 ani).

```
-- pentru a testa am inserat niste instante expirate
INSERT INTO instances
VALUES (200, DATE'2000-01-01');
INSERT INTO instances
VALUES (201, DATE'2001-01-01');
INSERT INTO participates
VALUES (1, 17, 200);
```

```sql
INSERT INTO participates
VALUES (1, 19, 200);
INSERT INTO participates
VALUES (2, 18, 201);
INSERT INTO participates
VALUES (2, 17, 201);

CREATE OR REPLACE TRIGGER trg_10
    AFTER INSERT
    ON instances
DECLARE
BEGIN
    FOR i IN (SELECT *
              FROM instances
              WHERE EXTRACT(YEAR FROM SYSDATE) - EXTRACT(YEAR FROM create_date) > 5)
        LOOP
            dbms_output.put_line('Expired instance found.');
            DELETE FROM participates WHERE instance_id = i.instance_id;
            DELETE FROM instances WHERE instance_id = i.instance_id;
        END LOOP;
END;
/

-- declansare trigger
INSERT INTO instances
VALUES (203, DATE'2010-01-01');

-- stergere trigger
drop trigger trg_10;

SELECT *
FROM instances;
```

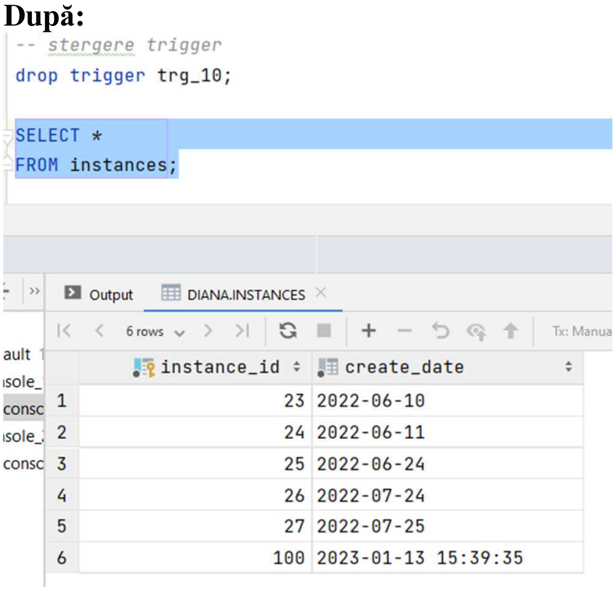
Înainte:



```sql
drop trigger trg_10;

SELECT *
FROM instances;
```

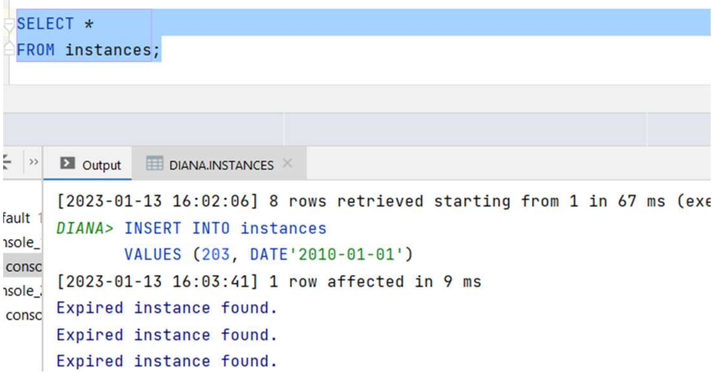
	instance_id	create_date
1	23	2022-06-10
2	24	2022-06-11
3	25	2022-06-24
4	26	2022-07-24
5	27	2022-07-25
6	100	2023-01-13 15:39:35
7	200	2000-01-01
8	201	2001-01-01

După:



```sql
-- stergere trigger
drop trigger trg_10;

SELECT *
FROM instances;
```

	instance_id	create_date
1	23	2022-06-10
2	24	2022-06-11
3	25	2022-06-24
4	26	2022-07-24
5	27	2022-07-25
6	100	2023-01-13 15:39:35



```sql
SELECT *
FROM instances;
```

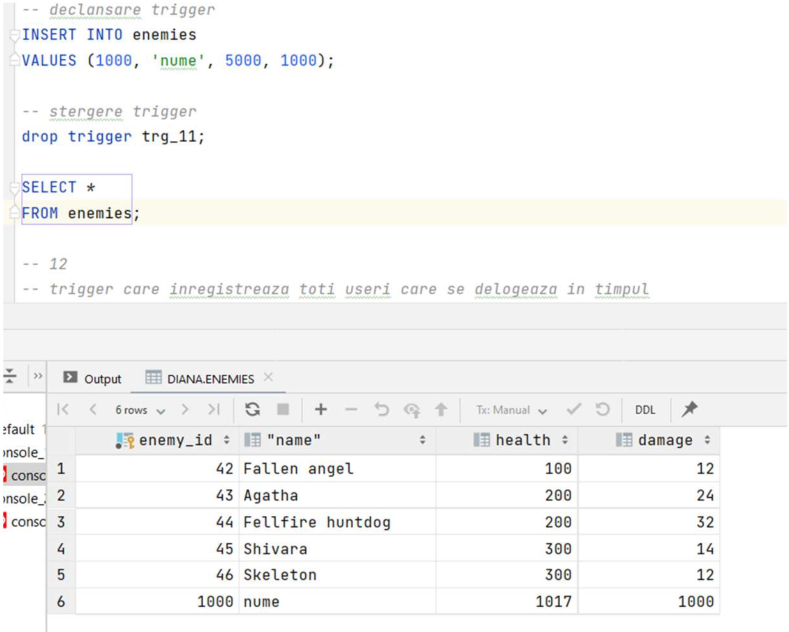
```
[2023-01-13 16:02:06] 8 rows retrieved starting from 1 in 67 ms (exe
DIANA> INSERT INTO instances
       VALUES (203, DATE'2010-01-01')
[2023-01-13 16:03:41] 1 row affected in 9 ms
Expired instance found.
Expired instance found.
Expired instance found.
```

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

Definesc un trigger care nu mă va lăsa să atribui unui inamic health mai mic decât o treime din medie sau health mai mare decat triplul medie.

- Trigger pe tabel mutatuing

```
-- compound trigger
CREATE OR REPLACE TRIGGER trg_11
    FOR UPDATE OR INSERT
    ON enemies
    COMPOUND TRIGGER
    TYPE rec IS RECORD
                (
                    enemy_id enemies.enemy_id%TYPE,
                    health   enemies.health%TYPE
                );

    TYPE ind_tb IS TABLE OF rec INDEX BY PLS_INTEGER;
    tab ind_tb;

AFTER EACH ROW IS
BEGIN
    tab(tab.count + 1).enemy_id := :new.enemy_id;
    tab(tab.count).health := :new.health;
END AFTER EACH ROW;

    AFTER STATEMENT IS
        avgi NUMBER;
    BEGIN
        SELECT AVG(health)
        INTO avgi
        FROM enemies
        WHERE health > 0;

        FOR i IN tab.first..tab.last
            LOOP
                IF tab(i).health < avgi / 3 OR tab(i).health > avgi * 3
                THEN
                    UPDATE enemies
                    SET health = avgi -- media rotunjita
                    WHERE enemies.enemy_id = tab(i).enemy_id;
                END IF;
            END LOOP;
    END AFTER STATEMENT;
    END trg_11;

-- declansare trigger
INSERT INTO enemies
VALUES (1000, 'nume', 5000, 1000);

-- stergere trigger
drop trigger trg_11;
```



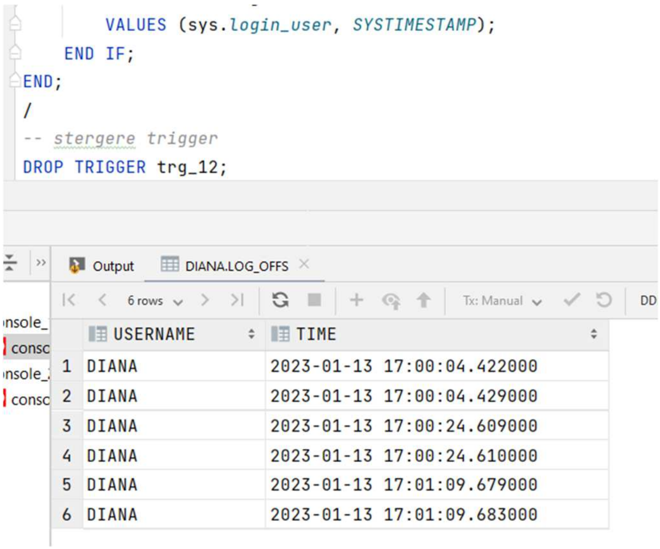
```
-- declansare trigger
INSERT INTO enemies
VALUES (1000, 'nume', 5000, 1000);

-- stergere trigger
drop trigger trg_11;

SELECT *
FROM enemies;

-- 12
-- trigger care inregistreaza toti useri care se delogeaza in timpul
```

	enemy_id	"name"	health	damage
1	42	Fallen angel	100	12
2	43	Agatha	200	24
3	44	Fellfire huntdog	200	32
4	45	Shivara	300	14
5	46	Skeleton	300	12
6	1000	nume	1017	1000

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

Definesc un trigger care reține în tabelul log_offs userii care s-au delogat în timpul programului.

```
CREATE OR REPLACE TRIGGER trg_12
    BEFORE LOGOFF
    ON SCHEMA
BEGIN
    IF TO_CHAR(SYSDATE, 'HH24') BETWEEN 6 AND 18 THEN
        INSERT INTO log_offs
        VALUES (sys.login_user, SYSTIMESTAMP);
    END IF;
END;
/
-- stergere trigger
DROP TRIGGER trg_12;

SELECT *
FROM log_offs;
```



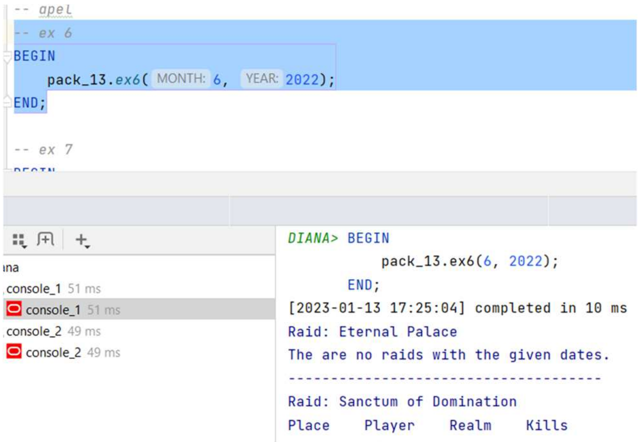
```
        VALUES (sys.login_user, SYSTIMESTAMP);
    END IF;
END;
/
-- stergere trigger
DROP TRIGGER trg_12;
```

	USERNAME	TIME
1	DIANA	2023-01-13 17:00:04.422000
2	DIANA	2023-01-13 17:00:04.429000
3	DIANA	2023-01-13 17:00:24.609000
4	DIANA	2023-01-13 17:00:24.610000
5	DIANA	2023-01-13 17:01:09.679000
6	DIANA	2023-01-13 17:01:09.683000

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

```
CREATE OR REPLACE PACKAGE pack_13 IS
    TYPE vect_b IS VARRAY(5000) OF NUMBER(38);
    TYPE vect_r IS TABLE OF VARCHAR(255);
    TYPE tabel IS TABLE OF vect_r;

    PROCEDURE ex6(month NUMBER, year NUMBER);
    PROCEDURE ex7(realm_name players.realm%TYPE);
    FUNCTION ex8(bosses_vect vect_b)
        RETURN tabel;
    PROCEDURE ex9(input_name players."name"%TYPE, output OUT NUMBER);
END pack_13;

CREATE OR REPLACE PACKAGE BODY pack_13 IS
    PROCEDURE ex6(month NUMBER, year NUMBER)
        IS
        TYPE player_rec IS RECORD
                           (
                               name  players."name"%TYPE,
                               realm players.realm%TYPE,
                               kills NUMBER
                           );
        TYPE vect_players IS VARRAY(1000) OF player_rec;
        TYPE raid_ranking IS RECORD
                             (
                                 raid_name raids."name"%TYPE,
                                 players   vect_players
                             );
        -- colectie 1 : colectie imbricata care contine numele raidului si lista de jucatori a
acelui raid
        TYPE rankings_tbl IS TABLE OF raid_ranking
            INDEX BY BINARY_INTEGER;
        rankings  rankings_tbl;
        v_players vect_players;
        -- colectie 2: vector de raiduri
        TYPE raids_vect IS VARRAY(100) OF raids%ROWTYPE;
        v_raids   raids_vect;
    BEGIN
        SELECT * BULK COLLECT
        INTO v_raids
        FROM raids;

        FOR i IN v_raids.first..v_raids.last
            LOOP
                SELECT MAX(players."name"),
                       MAX(players.realm),
                       COUNT(*) BULK COLLECT
                INTO v_players
                FROM players
                         JOIN participates USING (player_id)
                         JOIN instances USING (instance_id)
                         JOIN raids USING (raid_id)
                         JOIN maps USING (raid_id)
                         JOIN bosses USING (boss_id)
                WHERE raid_id = v_raids(i).raid_id
                  AND EXTRACT(MONTH FROM instances.create_date) = month
                  AND EXTRACT(YEAR FROM instances.create_date) = year
                GROUP BY player_id
                ORDER BY COUNT(*) DESC;

                rankings(i).raid_name := v_raids(i)."name";
                rankings(i).players := v_players;
            END LOOP;

        FOR i IN rankings.first..rankings.last
            LOOP
                dbms_output.put_line('Raid: ' || rankings(i).raid_name);

                IF rankings(i).players.count = 0 THEN
                    dbms_output.put_line('The are no raids with the given dates.');
                ELSE
                    dbms_output.put_line('Place    Player    Realm    Kills');
                    FOR j IN rankings(i).players.first..rankings(i).players.last
                        LOOP
                            dbms_output.put_line(j || CHR(9) || CHR(9) ||
rankings(i).players(j).name || CHR(9) ||
                                                 CHR(9) ||
                                                 rankings(i).players(j).realm || CHR(9) ||
CHR(9) ||

                                                 rankings(i).players(j).kills);
                        END LOOP;
                END IF;
                dbms_output.put_line('-------------------------------------');
            END LOOP;
    END ex6;

    PROCEDURE ex7(realm_name players.realm%TYPE)
        IS
        TYPE refcursor IS REF CURSOR;
        v_cursor refcursor;
        -- cursor 1 : cursor imbricat cu paramentru
        CURSOR c(realm_name players.realm%TYPE) IS
            (SELECT extern."name",
                    CURSOR (SELECT MAX("name"), COUNT(*)
                            FROM players
                                     JOIN participates USING (player_id)
                            WHERE LOWER(realm) LIKE LOWER(realm_name)
                              AND participates.raid_id = extern.raid_id
                            GROUP BY player_id
                            HAVING COUNT(*) = (SELECT MAX(COUNT(*))
                                               FROM players
                                                        JOIN participates USING (player_id)
                                               WHERE LOWER(realm) LIKE LOWER(realm_name)
                                                 AND participates.raid_id = extern.raid_id
                                               GROUP BY player_id))
             FROM raids extern);
        v_raid   raids."name"%TYPE;
        v_nr     NUMBER;
        v_name   players."name"%TYPE;
        found    NUMBER := 0;
        -- cursor 2 :
        CURSOR c2 IS (SELECT DISTINCT realm
                      FROM players);
        error    players.realm%TYPE;
        no_realm EXCEPTION;
        PRAGMA EXCEPTION_INIT (no_realm, -9090);
    BEGIN
        SELECT COUNT(*)
        INTO error
        FROM players
        WHERE LOWER(realm) LIKE LOWER(realm_name);

        IF error = 0 THEN
            RAISE no_realm;
        END IF;

        OPEN c(realm_name);
        LOOP
            FETCH c INTO v_raid, v_cursor;
            EXIT WHEN c%NOTFOUND;
            dbms_output.put_line('-------------------------------');
            dbms_output.put_line('RAID ' || UPPER(v_raid));
            dbms_output.put_line('-------------------------------');
            found := 0;
            LOOP
                FETCH v_cursor INTO v_name, v_nr;
                EXIT WHEN v_cursor%NOTFOUND;
                found := 1;
                dbms_output.put_line(v_name || ' total raids: ' || v_nr);
            END LOOP;
            IF found = 0 THEN
                dbms_output.put_line('None.');
            END IF;
        END LOOP;
        CLOSE c;
    EXCEPTION
        WHEN no_realm THEN
            dbms_output.put_line('No realm with name: ' || realm_name);
            dbms_output.put_line('Reaml list: ');
            FOR r IN c2
                LOOP
                    dbms_output.put_line(r.realm);
                END LOOP;
    END ex7;

    FUNCTION ex8(bosses_vect vect_b)
        RETURN tabel IS

        -- 3 exceptii
        boss_not_found EXCEPTION;
        PRAGMA EXCEPTION_INIT (boss_not_found, -2000);

        emply_vector EXCEPTION;
        PRAGMA EXCEPTION_INIT (emply_vector, -3000);

        null_vector EXCEPTION;
        PRAGMA EXCEPTION_INIT (null_vector, -4000);
        nr     NUMBER;
        maxi   bosses.boss_id%TYPE;
        vector vect_r := vect_r();
        output tabel  := tabel();
    BEGIN
        IF bosses_vect IS NULL THEN
            RAISE null_vector;
        END IF;

        IF bosses_vect.count = 0 THEN
            RAISE emply_vector;
        END IF;

        -- !!!!!!
        SELECT MAX(boss_id)
        INTO maxi
        FROM bosses;

        FOR i IN 1..maxi
            LOOP
```

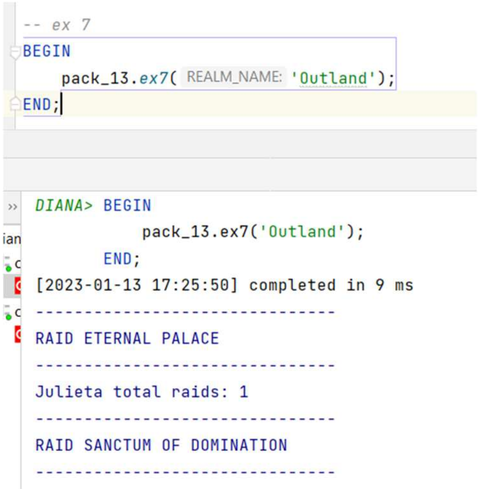
```
                output.extend;
            END LOOP;

        FOR i IN bosses_vect.first..bosses_vect.last
            LOOP
                SELECT COUNT(*)
                INTO nr
                FROM bosses
                WHERE boss_id = bosses_vect(i);

                IF nr = 0 THEN
                    RAISE boss_not_found;
                END IF;

                -- comanda cu 3 tabele
                SELECT DISTINCT rewards."name" BULK COLLECT
                INTO vector
                FROM rewards
                         JOIN has ON has.reward_id = rewards.reward_id
                         JOIN bonuses ON bonuses.bonus_id = has.bonus_id
                WHERE bonuses.quantity > ALL (SELECT bonuses.quantity
                                              FROM rewards
                                                       JOIN has ON has.reward_id =
rewards.reward_id
                                                       JOIN bonuses ON bonuses.bonus_id =
has.bonus_id
                                              WHERE rewards.boss_id = bosses_vect(i));
                IF vector IS NULL THEN
                    output(bosses_vect(i)) := vect_r();
                ELSE
                    output(bosses_vect(i)) := vector;
                END IF;
            END LOOP;
        RETURN output;
    EXCEPTION
        WHEN emply_vector THEN
            dbms_output.put_line('Vector is empty.');
            RETURN NULL;
        WHEN null_vector THEN
            dbms_output.put_line('Vector is null.');
            RETURN NULL;
        WHEN boss_not_found THEN
            dbms_output.put_line('No boss found.');
            RETURN NULL;
    END ex8;

    PROCEDURE ex9(input_name players."name"%TYPE, output OUT NUMBER)
        IS
        id       players.player_id%TYPE;
        raids_nr NUMBER;
        v_sum    NUMBER;
        no_rewards EXCEPTION;
        PRAGMA EXCEPTION_INIT (no_rewards, -3000);
    BEGIN
        SELECT player_id
        INTO id
        FROM players
        WHERE "name" = input_name;

        SELECT COUNT(*)
        INTO raids_nr
        FROM participates
        WHERE player_id = id;

        SELECT SUM(NVL(sell_price, 0))
        INTO v_sum
        FROM participates
                 JOIN raids USING (raid_id)
                 JOIN maps USING (raid_id)
                 JOIN bosses USING (boss_id)
                 JOIN rewards USING (boss_id)
        WHERE player_id = id;

        output := NVL(v_sum, 0) / raids_nr;

        IF v_sum IS NULL THEN
            RAISE no_rewards;
        END IF;

    EXCEPTION
        WHEN too_many_rows THEN
            dbms_output.put_line('More players with name: ' || input_name);
        WHEN no_data_found THEN
            dbms_output.put_line('No player with name: ' || input_name);
        WHEN no_rewards THEN
            dbms_output.put_line('Player ' || input_name || ' has no rewards');
        WHEN zero_divide THEN
            dbms_output.put_line('Player ' || input_name || ' has not participated to any raid
yet');
    END ex9;
END pack_13;

-- ex 6
BEGIN
    pack_13.ex6(6, 2022);
END;
```



```
-- apel
-- ex 6
BEGIN
    pack_13.ex6( MONTH: 6,  YEAR: 2022);
END;

-- ex 7
```

```
DIANA> BEGIN
            pack_13.ex6(6, 2022);
        END;
[2023-01-13 17:25:04] completed in 10 ms
Raid: Eternal Palace
The are no raids with the given dates.
-------------------------------------
Raid: Sanctum of Domination
Place     Player     Realm     Kills
```

```
-- ex 7
BEGIN
    pack_13.ex7('Outland');
END;
```

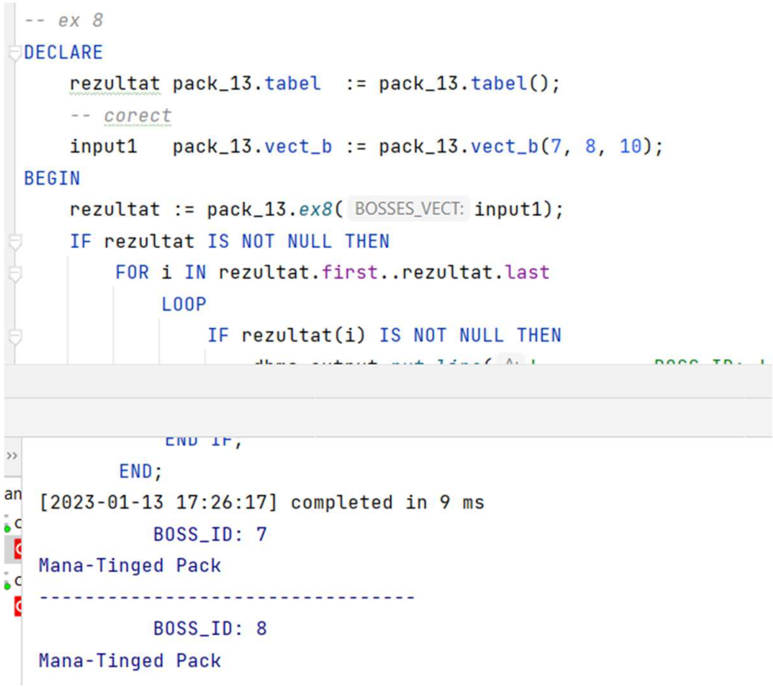


```
-- ex 7
BEGIN
    pack_13.ex7( REALM_NAME: 'Outland');
END;
```

```
DIANA> BEGIN
            pack_13.ex7('Outland');
        END;
[2023-01-13 17:25:50] completed in 9 ms
-------------------------------
RAID ETERNAL PALACE
-------------------------------
Julieta total raids: 1
-------------------------------
RAID SANCTUM OF DOMINATION
-------------------------------
```

```
-- ex 8
DECLARE
    rezultat pack_13.tabel  := pack_13.tabel();
    -- corect
    input1   pack_13.vect_b := pack_13.vect_b(7, 8, 10);
BEGIN
    rezultat := pack_13.ex8(input1);
    IF rezultat IS NOT NULL THEN
        FOR i IN rezultat.first..rezultat.last
            LOOP
                IF rezultat(i) IS NOT NULL THEN
                    dbms_output.put_line('          BOSS_ID: ' || i || '           ');

                    IF rezultat(i).count != 0 THEN
                        FOR j IN rezultat(i).first..rezultat(i).last
```

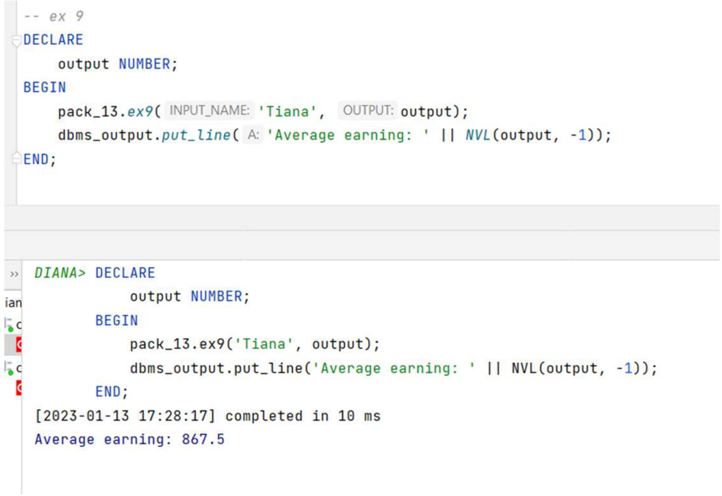
```
                            LOOP
                                dbms_output.put_line(rezultat(i)(j));
                            END LOOP;
                    ELSE
                        dbms_output.put_line('None.');
                    END IF;
                    dbms_output.put_line('---------------------------------');
                END IF;
            END LOOP;
    END IF;
END;
```



```
-- ex 8
DECLARE
    rezultat pack_13.tabel  := pack_13.tabel();
    -- corect
    input1   pack_13.vect_b := pack_13.vect_b(7, 8, 10);
BEGIN
    rezultat := pack_13.ex8( BOSSES_VECT: input1);
    IF rezultat IS NOT NULL THEN
        FOR i IN rezultat.first..rezultat.last
            LOOP
                IF rezultat(i) IS NOT NULL THEN
                    dbms_output.put_line( A: '          BOSS_ID: '
```

```
            END IF;
        END;
[2023-01-13 17:26:17] completed in 9 ms
          BOSS_ID: 7
Mana-Tinged Pack
---------------------------------
          BOSS_ID: 8
Mana-Tinged Pack
```

```
-- ex 9
DECLARE
    output NUMBER;
BEGIN
    pack_13.ex9('Tiana', output);
    dbms_output.put_line('Average earning: ' || NVL(output, -1));
END;
```



```
-- ex 9
DECLARE
    output NUMBER;
BEGIN
    pack_13.ex9( INPUT_NAME: 'Tiana',  OUTPUT: output);
    dbms_output.put_line( A: 'Average earning: ' || NVL(output, -1));
END;
```

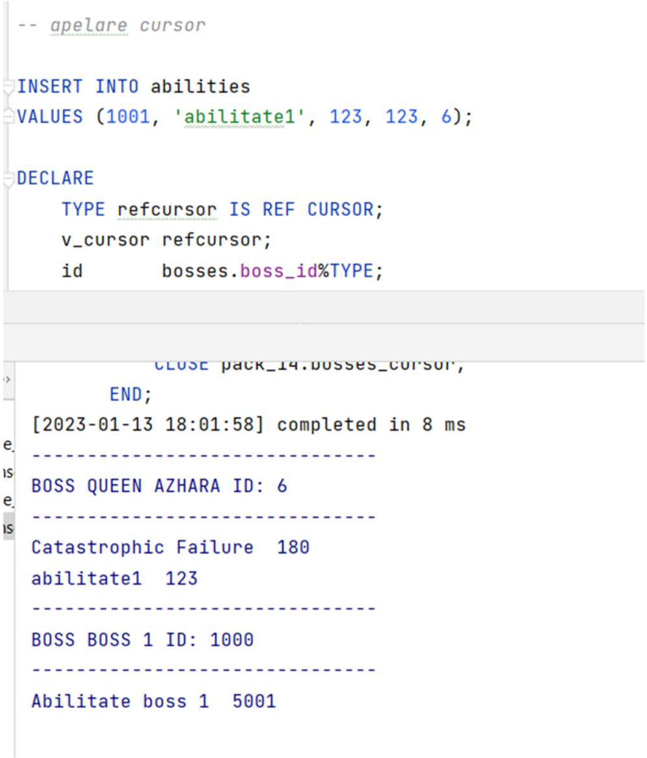
```
DIANA> DECLARE
           output NUMBER;
       BEGIN
           pack_13.ex9('Tiana', output);
           dbms_output.put_line('Average earning: ' || NVL(output, -1));
       END;
[2023-01-13 17:28:17] completed in 10 ms
Average earning: 867.5
```

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).

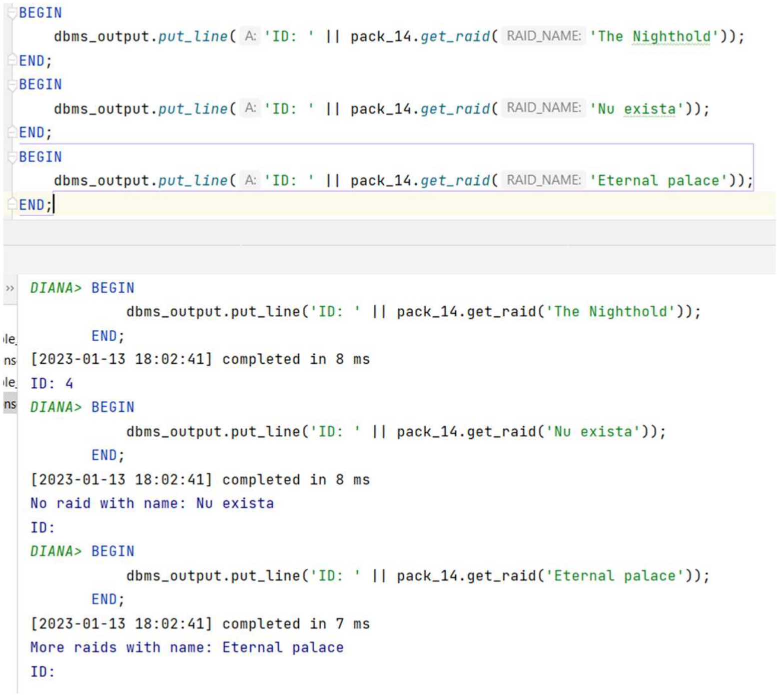
```
-- adaug nou boss la raid-ul "Enternal Palace"
INSERT INTO bosses
VALUES (1000, 'Boss 1', 100, 1000, 100, NULL);

INSERT INTO maps
VALUES (1000, 'harta boss 1', 1, 1000);

INSERT INTO abilities
VALUES (1000, 'Abilitate boss 1', 100, 5001, 1000);

CREATE OR REPLACE PACKAGE pack_14 IS
    -- cursor imbricat cu parametru care pentru un raid dat
    -- selecteaza bossi si abilitatile lor
    CURSOR bosses_cursor(raid_id_input raids.raid_id%TYPE) IS
        (SELECT extern.boss_id,
                extern."name",
                CURSOR (SELECT *
                        FROM abilities
                        WHERE boss_id = extern.boss_id)
         FROM bosses extern
                  JOIN maps ON extern.boss_id = maps.boss_id
         WHERE raid_id = raid_id_input);
    -- functie care returneaza id-ul unui raid fiind dat ca param numele sau
    FUNCTION get_raid(raid_name raids."name"%TYPE)
        RETURN raids.raid_id%TYPE;
    --pentru fiecare boss din raidul dat sa ii se calculeze dificultatea (in functie de cat de
puternice sunt abilitatile sale)
    PROCEDURE calc_dif(raid_name raids."name"%TYPE);


    TYPE player_list_type IS VARRAY(1000) OF players%ROWTYPE;
    TYPE team_rec IS RECORD
                     (
                         raid_name   raids."name"%TYPE,
                         raid_id     raids.raid_id%TYPE,
                         player_list player_list_type
                     );
    -- tab indexat de vectori care retine echipele de jucatori si raidul la care au participat
    TYPE teams_type IS TABLE OF team_rec
        INDEX BY PLS_INTEGER;
    teams teams_type;
    FUNCTION get_reward_money(idd raids.raid_id%TYPE)
        RETURN NUMBER;
    PROCEDURE calc_teams;
    PROCEDURE print_teams;

END pack_14;

CREATE OR REPLACE PACKAGE BODY pack_14 IS
    -- functie care returneaza id-ul unui raid fiind dat ca param numele sau
    FUNCTION get_raid(raid_name raids."name"%TYPE)
        RETURN raids.raid_id%TYPE IS
        id raids.raid_id%TYPE;
    BEGIN
        SELECT raid_id
        INTO id
        FROM raids
        WHERE LOWER("name") LIKE LOWER(raid_name);
        RETURN id;
    EXCEPTION
        WHEN no_data_found THEN
            dbms_output.put_line('No raid with name: ' || raid_name);
            RETURN NULL;
        WHEN too_many_rows THEN
            dbms_output.put_line('More raids with name: ' || raid_name);
            RETURN NULL;
    END get_raid;

    --pentru fiecare boss din raidul dat sa ii se calculeze dificultatea (in functie de cat de
puternice sunt abilitatile sale)
    -- dc sum(damage) < 200 => easy
    --    200 <= sum(damage) < 400 => medium
    --    altfel => hard
    PROCEDURE calc_dif(raid_name raids."name"%TYPE) IS
        id       raids.raid_id%TYPE;
        id_boss  bosses.boss_id%TYPE;
        name     bosses."name"%TYPE;
        TYPE refcursor IS REF CURSOR;
        v_cursor refcursor;
        summ     NUMBER;
        dific    VARCHAR(255);
        ability  abilities%ROWTYPE;
    BEGIN
        id := get_raid(raid_name);
        IF id IS NOT NULL THEN
            OPEN bosses_cursor(id);
            LOOP
                FETCH bosses_cursor INTO id_boss, name, v_cursor;
                EXIT WHEN bosses_cursor%NOTFOUND;
                summ := 0;
                LOOP
                    FETCH v_cursor INTO ability;
                    EXIT WHEN v_cursor%NOTFOUND;
                    summ := summ + NVL(ability.damage, 0);
                END LOOP;
                CASE
                    WHEN summ < 200 THEN dific := 'Easy';
                    WHEN summ < 400 THEN dific := 'Medium';
                    ELSE dific := 'Hard';
                    END CASE;
                UPDATE bosses
                SET bosses.dificulty = dific
                WHERE boss_id = id_boss;
            END LOOP;
            CLOSE bosses_cursor;
        END IF;
    END calc_dif;

    -- procedura care afla fiecare echipa din baza de date
    -- (echipa = jucatori care au participat la aceeasi instanta)
    PROCEDURE calc_teams IS
        player_list player_list_type;
        raid_name   raids."name"%TYPE;
        idd         raids.raid_id%TYPE;
        nr          NUMBER := 0;
    BEGIN
        FOR inst IN (SELECT DISTINCT instance_id FROM participates)
            LOOP
                SELECT players.* BULK COLLECT
                INTO player_list
                FROM players
                         JOIN participates ON players.player_id = participates.player_id
                WHERE instance_id = inst.instance_id;

                SELECT DISTINCT "name", raid_id
                INTO raid_name, idd
                FROM participates
                         JOIN raids USING (raid_id)
                WHERE instance_id = inst.instance_id;

                nr := nr + 1;
                teams(nr).player_list := player_list;
                teams(nr).raid_name := raid_name;
                teams(nr).raid_id := idd;
            END LOOP;
    END calc_teams;

    -- functie care returneaza suma totala de bani castigata in cadrul raidului dat
    -- ca parametru
    FUNCTION get_reward_money(idd raids.raid_id%TYPE)
        RETURN NUMBER
        IS
        summ NUMBER := 0;
        err  raids.raid_id%TYPE;
    BEGIN
        SELECT raid_id
        INTO err
        FROM raids
        WHERE raid_id = idd;

        SELECT SUM(NVL(sell_price, 0))
```

```
        INTO summ
        FROM raids
                 JOIN maps USING (raid_id)
                 JOIN bosses USING (boss_id)
                 JOIN rewards USING (boss_id)
        WHERE raid_id = idd;
        RETURN summ;
    EXCEPTION
        WHEN no_data_found THEN
            dbms_output.put_line('No raid with id: ' || idd);
            RETURN NULL;
    END get_reward_money;


    -- procedura care afiseaza echipele impreuna cu suma de bani castigata
    PROCEDURE print_teams IS
    BEGIN
        calc_teams;
        FOR i IN teams.first..teams.last
            LOOP
                dbms_output.put_line('      TEAM ' || i || '      ');
                dbms_output.put_line('raid: ' || teams(i).raid_name);
                FOR p IN teams(i).player_list.first..teams(i).player_list.last
                    LOOP
                        dbms_output.put_line(teams(i).player_list(p).player_id || ' ' ||
                                             teams(i).player_list(p)."name");
                    END LOOP;
                dbms_output.put_line('money earned ' ||
NVL(get_reward_money(teams(i).raid_id), 0));
                dbms_output.put_line('-------------------------');
            END LOOP;
    END print_teams;
END pack_14;

-- apelare cursor

INSERT INTO abilities
VALUES (1001, 'abilitate1', 123, 123, 6);

DECLARE
    TYPE refcursor IS REF CURSOR;
    v_cursor refcursor;
    id       bosses.boss_id%TYPE;
    name     bosses."name"%TYPE;
    ability  abilities%ROWTYPE;
BEGIN
    OPEN pack_14.bosses_cursor(1);
    LOOP
        FETCH pack_14.bosses_cursor INTO id, name, v_cursor;
        EXIT WHEN pack_14.bosses_cursor%NOTFOUND;
        dbms_output.put_line('-------------------------------');
        dbms_output.put_line('BOSS ' || UPPER(name) || ' ID: ' || id);
        dbms_output.put_line('-------------------------------');
        LOOP
            FETCH v_cursor INTO ability;
            EXIT WHEN v_cursor%NOTFOUND;
            dbms_output.put_line(ability."name" || '  ' || ability.damage);
        END LOOP;
    END LOOP;
    CLOSE pack_14.bosses_cursor;
END;

-- apelare functie get_raid
INSERT INTO raids
VALUES (1000, 'Eternal palace', 'locatie', 12, 40);

BEGIN
    dbms_output.put_line('ID: ' || pack_14.get_raid('The Nighthold'));
END;
BEGIN
    dbms_output.put_line('ID: ' || pack_14.get_raid('Nu exista'));
END;
BEGIN
    dbms_output.put_line('ID: ' || pack_14.get_raid('Eternal palace'));
END;

-- apelare procedura calc_dif
```

```
INSERT INTO raids
VALUES (1001, 'raid fara boss-i', 'loc', 5, 60);

BEGIN
    pack_14.calc_dif('nu exista');
END;
BEGIN
    pack_14.calc_dif('Eternal palace');
END;
BEGIN
    pack_14.calc_dif('Icecrown Citadel');
END;
BEGIN
    pack_14.calc_dif('raid fara boss-i');
END;

-- apel proc calc_teams
BEGIN
    pack_14.print_teams;
END;
```

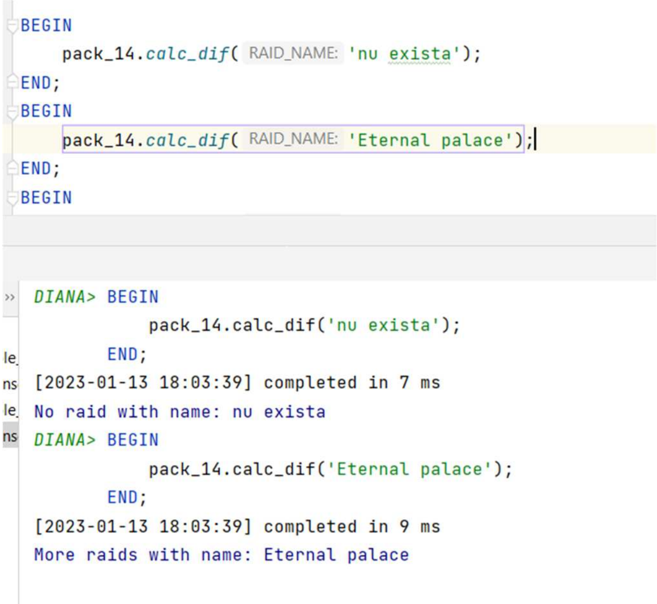


```
-- apelare cursor

INSERT INTO abilities
VALUES (1001, 'abilitate1', 123, 123, 6);

DECLARE
    TYPE refcursor IS REF CURSOR;
    v_cursor refcursor;
    id       bosses.boss_id%TYPE;

        CLOSE pack_14.bosses_cursor;
    END;
[2023-01-13 18:01:58] completed in 8 ms
-------------------------------
BOSS QUEEN AZHARA ID: 6
-------------------------------
Catastrophic Failure  180
abilitate1  123
-------------------------------
BOSS BOSS 1 ID: 1000
-------------------------------
Abilitate boss 1  5001
```



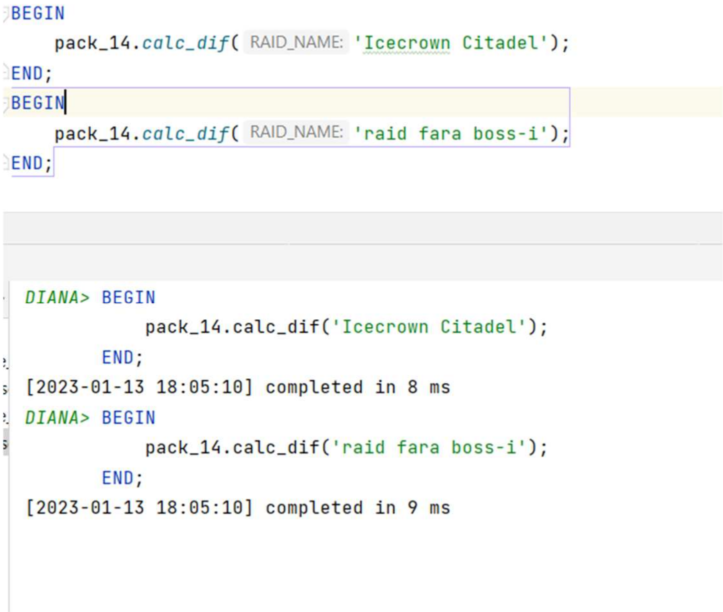
```
BEGIN
    dbms_output.put_line( A: 'ID: ' || pack_14.get_raid( RAID_NAME: 'The Nighthold'));
END;
BEGIN
    dbms_output.put_line( A: 'ID: ' || pack_14.get_raid( RAID_NAME: 'Nu exista'));
END;
BEGIN
    dbms_output.put_line( A: 'ID: ' || pack_14.get_raid( RAID_NAME: 'Eternal palace'));
END;
```

```
DIANA> BEGIN
           dbms_output.put_line('ID: ' || pack_14.get_raid('The Nighthold'));
       END;
[2023-01-13 18:02:41] completed in 8 ms
ID: 4
DIANA> BEGIN
           dbms_output.put_line('ID: ' || pack_14.get_raid('Nu exista'));
       END;
[2023-01-13 18:02:41] completed in 8 ms
No raid with name: Nu exista
ID:
DIANA> BEGIN
           dbms_output.put_line('ID: ' || pack_14.get_raid('Eternal palace'));
       END;
[2023-01-13 18:02:41] completed in 7 ms
More raids with name: Eternal palace
ID:
```



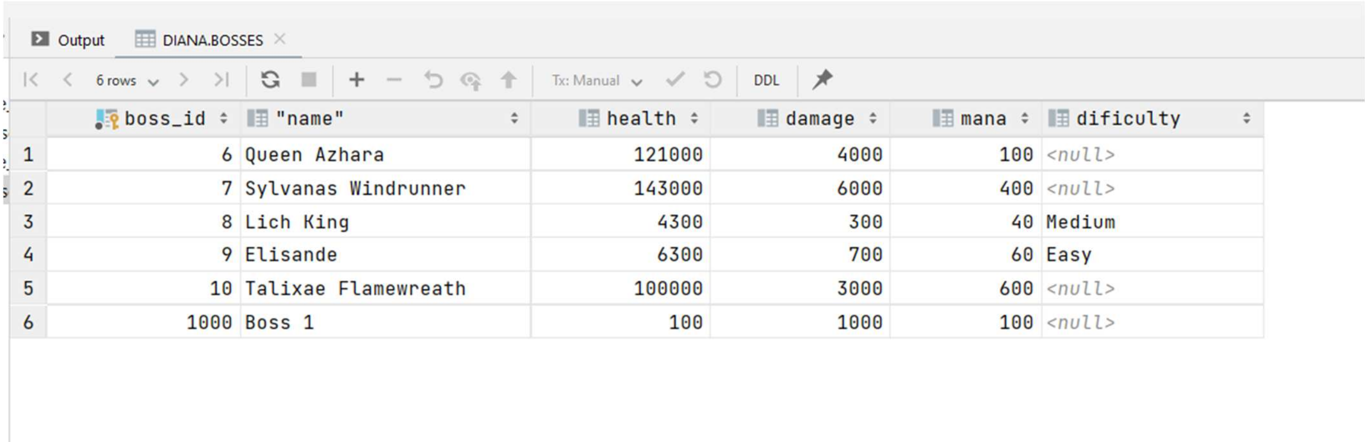
```
BEGIN
    pack_14.calc_dif( RAID_NAME: 'nu exista');
END;
BEGIN
    pack_14.calc_dif( RAID_NAME: 'Eternal palace');
END;
BEGIN
```

```
DIANA> BEGIN
           pack_14.calc_dif('nu exista');
       END;
[2023-01-13 18:03:39] completed in 7 ms
No raid with name: nu exista
DIANA> BEGIN
           pack_14.calc_dif('Eternal palace');
       END;
[2023-01-13 18:03:39] completed in 9 ms
More raids with name: Eternal palace
```

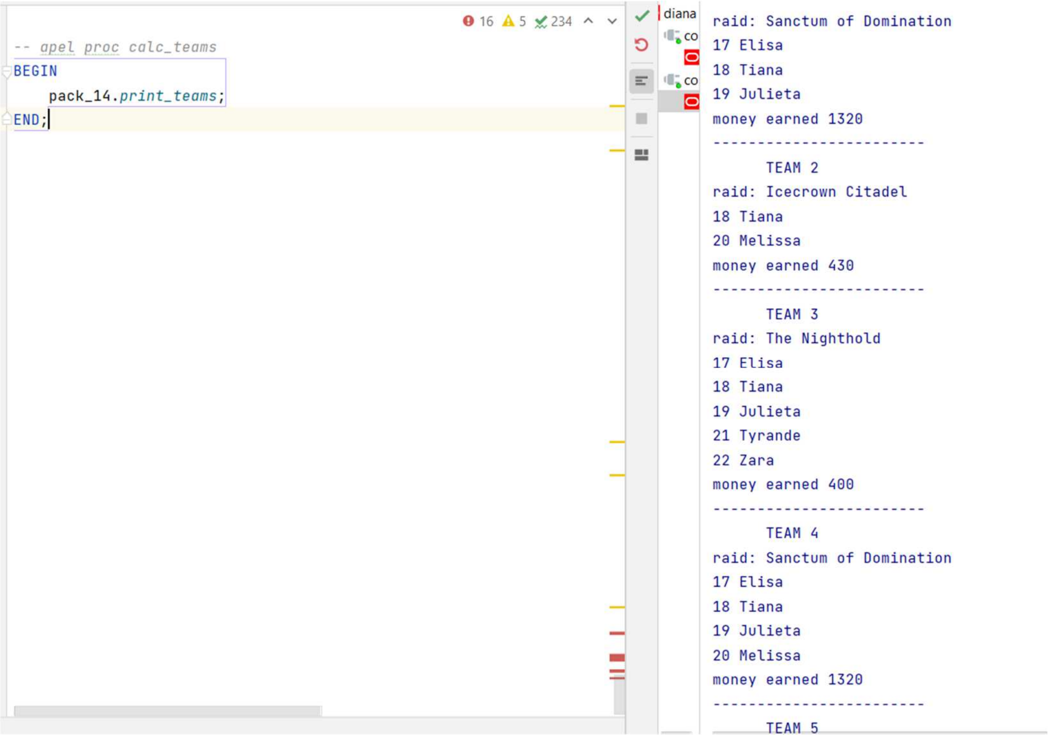


```
BEGIN
    pack_14.calc_dif( RAID_NAME: 'Icecrown Citadel');
END;
BEGIN
    pack_14.calc_dif( RAID_NAME: 'raid fara boss-i');
END;
```

```
DIANA> BEGIN
           pack_14.calc_dif('Icecrown Citadel');
       END;
[2023-01-13 18:05:10] completed in 8 ms
DIANA> BEGIN
           pack_14.calc_dif('raid fara boss-i');
       END;
[2023-01-13 18:05:10] completed in 9 ms
```



	boss_id	"name"	health	damage	mana	dificulty
1	6	Queen Azhara	121000	4000	100	<null>
2	7	Sylvanas Windrunner	143000	6000	400	<null>
3	8	Lich King	4300	300	40	Medium
4	9	Elisande	6300	700	60	Easy
5	10	Talixae Flamewreath	100000	3000	600	<null>
6	1000	Boss 1	100	1000	100	<null>



```
-- apel proc calc_teams
BEGIN
    pack_14.print_teams;
END;
```

```
raid: Sanctum of Domination
17 Elisa
18 Tiana
19 Julieta
money earned 1320
------------------------
      TEAM 2
raid: Icecrown Citadel
18 Tiana
20 Melissa
money earned 430
------------------------
      TEAM 3
raid: The Nighthold
17 Elisa
18 Tiana
19 Julieta
21 Tyrande
22 Zara
money earned 400
------------------------
      TEAM 4
raid: Sanctum of Domination
17 Elisa
18 Tiana
19 Julieta
20 Melissa
money earned 1320
------------------------
      TEAM 5
```