

Proiect SGBD

NUME: Negulescu Stefan

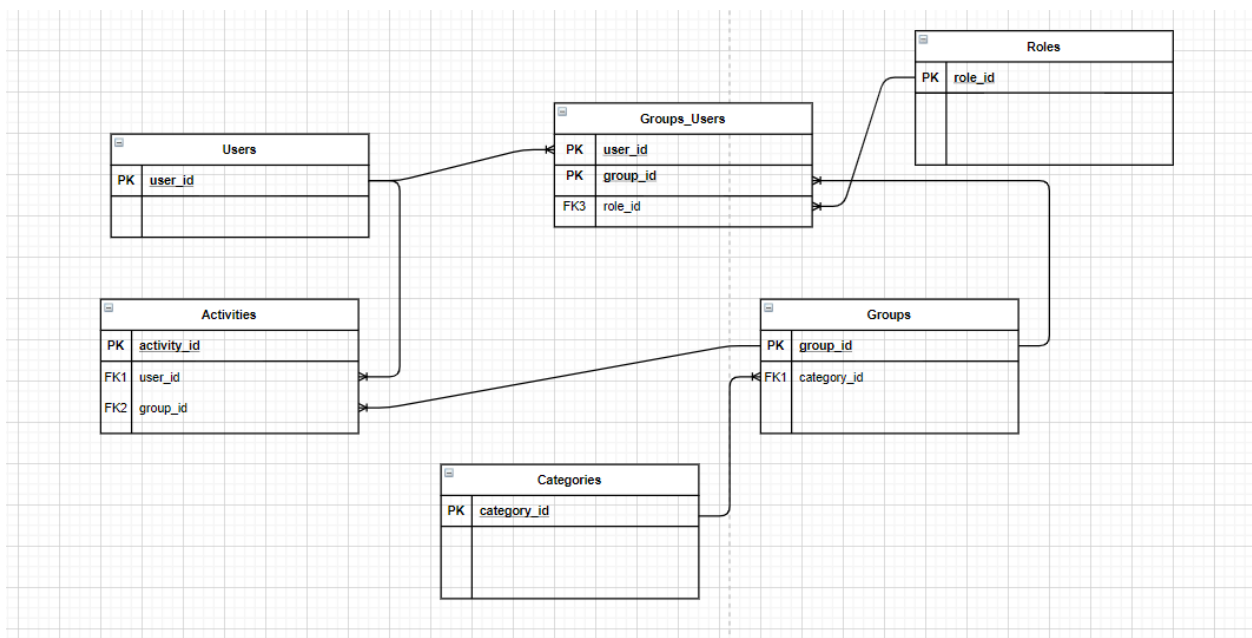
GRUPA: 232

1)

Acest model de baza de date poate apartine unei aplicatii bazate pe grupuri. Astfel un utilizator poate fi in mai multe grupuri, avand diferite roluri in functie de grup, iar un grup poate avea mai multi membrii. Un grup apartine unei anumite categorii. De asemenea, un utilizator poate introduce intr-un grup mai multe activitati.

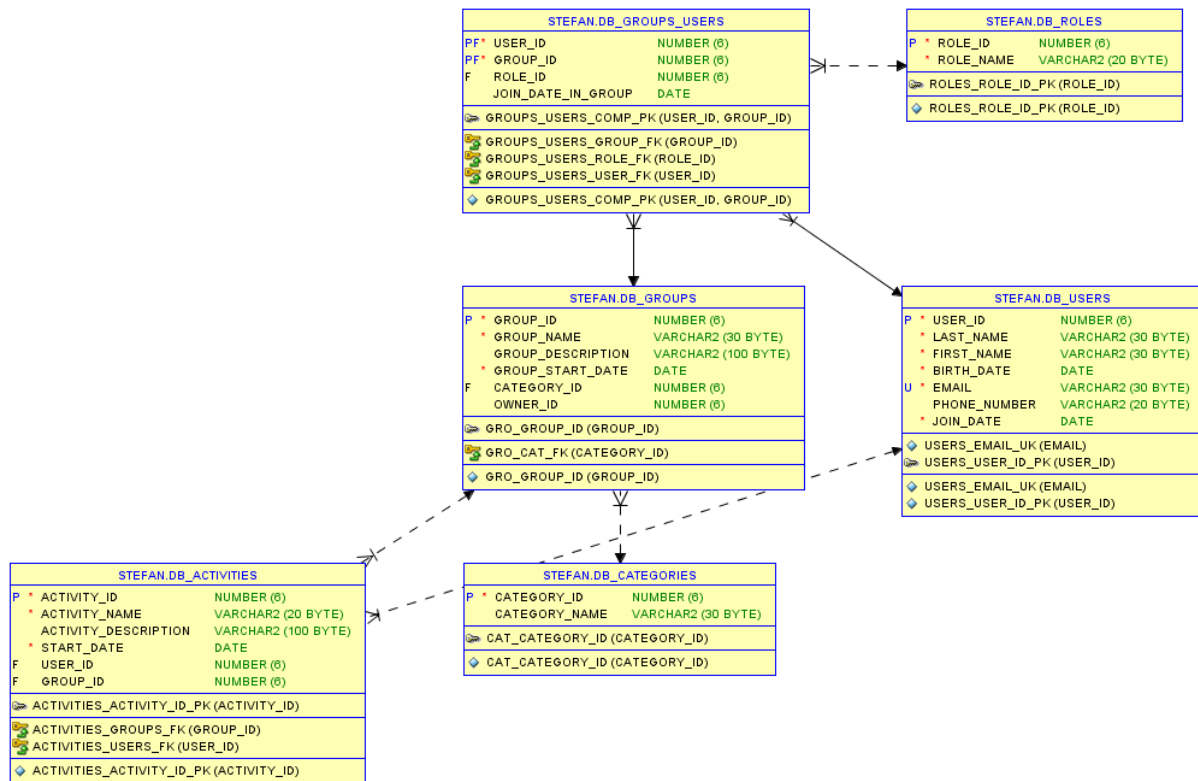
2)

Diagrama entitate-relatie



3)

Diagrama conceptuala



4)

-- 4

```

CREATE TABLE db_categories (
    category_id NUMBER(6,0),
    category_name VARCHAR2(30),
    CONSTRAINT cat_category_id PRIMARY KEY (category_id)
);
  
```

```
CREATE TABLE db_groups (  
    group_id NUMBER(6,0),  
    group_name VARCHAR2(30) NOT NULL,  
    group_description VARCHAR2(100),  
    group_start_date DATE NOT NULL,  
    category_id NUMBER(6,0),  
    owner_id NUMBER(6,0),  
    CONSTRAINT gro_group_id PRIMARY KEY (group_id),  
    CONSTRAINT gro_cat_fk FOREIGN KEY (category_id)  
        REFERENCES db_categories(category_id)  
);
```

```
CREATE TABLE db_users (  
    user_id NUMBER(6,0),  
    last_name VARCHAR2(30) NOT NULL,  
    first_name VARCHAR2(30) NOT NULL,  
    birth_date DATE NOT NULL,  
    email VARCHAR2(30) NOT NULL,  
    phone_number VARCHAR2(20),  
    join_date DATE NOT NULL,  
    CONSTRAINT users_email_uk UNIQUE (email),  
    CONSTRAINT users_user_id_pk PRIMARY KEY (user_id)  
);
```

```
CREATE TABLE db_roles (  
    role_id NUMBER(6,0),  
    role_name VARCHAR2(20) NOT NULL,  
    CONSTRAINT roles_role_id_pk PRIMARY KEY (role_id)  
);
```

```
CREATE TABLE db_groups_users (  
    user_id NUMBER(6,0),  
    group_id NUMBER(6,0),  
    role_id NUMBER(6,0),  
    join_date_in_group DATE,  
    CONSTRAINT groups_users_comp_pk PRIMARY KEY (user_id, group_id),  
    CONSTRAINT groups_users_user_fk FOREIGN KEY (user_id)  
        REFERENCES db_users(user_id),  
    CONSTRAINT groups_users_group_fk FOREIGN KEY (group_id)  
        REFERENCES db_groups(group_id),  
    CONSTRAINT groups_users_role_fk FOREIGN KEY (role_id)  
        REFERENCES db_roles(role_id)  
);
```

```

CREATE TABLE db_activities (
    activity_id NUMBER(6,0),
    activity_name VARCHAR2(20) NOT NULL,
    activity_description VARCHAR2(100),
    start_date DATE NOT NULL,
    user_id NUMBER(6,0),
    group_id NUMBER(6,0),
    CONSTRAINT activities_activity_id_pk PRIMARY KEY (activity_id),
    CONSTRAINT activities_users_fk FOREIGN KEY (user_id)
        REFERENCES db_users(user_id),
    CONSTRAINT activities_groups_fk FOREIGN KEY (group_id)
        REFERENCES db_groups(group_id)
);

```

5)

-- pentru categorii

```
CREATE SEQUENCE sec_cat
```

```
START WITH 1
```

```
MAXVALUE 9999;
```

-- pentru useri

```
CREATE SEQUENCE sec_usr
```

```
START WITH 1
```

```
MAXVALUE 9999;
```

```

-- pentru grupuri
CREATE SEQUENCE sec_gro
START WITH 1
MAXVALUE 9999;

-- pentru activitati
CREATE SEQUENCE sec_act
START WITH 1
MAXVALUE 9999;

-- pentru roluri
CREATE SEQUENCE sec_rol
START WITH 1
MAXVALUE 9999;

-- 5
BEGIN

--INSERT INTO db_users
--VALUES (sec_usr.nextval,'Petersen','Jordan',TO_DATE('11-03-1973','dd-mm-
yyyy'),'jpeterson@gmail.com','0722634812',
--TO_DATE('11-03-2010','dd-mm-yyyy'));

INSERT INTO db_users
VALUES (sec_usr.nextval,'Muller','Thomas',TO_DATE('12-04-1983','dd-mm-
yyyy'),'tmuller@gmail.com','0722004811',
TO_DATE('11-03-2010','dd-mm-yyyy'));

INSERT INTO db_users

```

```

VALUES (sec_usr.nextval,'Rodriguez','James',TO_DATE('24-11-1985','dd-mm-
yyyy'),'jrodriguez@gmail.com','0799634002',
TO_DATE('22-09-2011','dd-mm-yyyy'));

INSERT INTO db_users

VALUES (sec_usr.nextval,'Thompson','Steven',TO_DATE('02-03-1978','dd-mm-
yyyy'),'tsteven@gmail.com','0787654123',
TO_DATE('29-03-2011','dd-mm-yyyy'));

INSERT INTO db_users

VALUES (sec_usr.nextval,'Wilson','Jack',TO_DATE('24-12-1990','dd-mm-
yyyy'),'jwilson@gmail.com','0798547123',
TO_DATE('21-06-2010','dd-mm-yyyy'));

INSERT INTO db_users

VALUES (sec_usr.nextval,'Maxwell','Bernard',TO_DATE('09-08-1988','dd-mm-
yyyy'),'bmaxwell@gmail.com','0772345671',
TO_DATE('22-08-2013','dd-mm-yyyy'));

INSERT INTO db_users

VALUES (sec_usr.nextval,'Henderson','Jordan',TO_DATE('17-06-1986','dd-mm-
yyyy'),'jhenderson@gmail.com','0788654283',
TO_DATE('14-05-2012','dd-mm-yyyy'));

INSERT INTO db_users

VALUES (sec_usr.nextval,'Stones','John',TO_DATE('17-12-1987','dd-mm-
yyyy'),'jstones.com','0721333019',
TO_DATE('14-05-2012','dd-mm-yyyy'));

END;

/

SELECT *
```


FROM db_users;

	USER_ID	LAST_NAME	FIRST_NAME	BIRTH_DATE	EMAIL	PHONE_NUMBER	JOIN_DATE
1	1	Petersen	Jordan	11-MAR-73	jpetersen@gmail.com	0722634812	11-MAR-10
2	2	Muller	Thomas	12-APR-83	tmuller@gmail.com	0722004811	11-MAR-10
3	3	Rodriguez	James	24-NOV-85	jrodriguez@gmail.com	0799634002	22-SEP-11
4	4	Thompson	Steven	02-MAR-78	tsteven@gmail.com	0787654123	29-MAR-11
5	5	Wilson	Jack	24-DEC-90	jwilson@gmail.com	0798547123	21-JUN-10
6	6	Maxwell	Bernard	09-AUG-88	bmaxwell@gmail.com	0772345671	22-AUG-13
7	7	Henderson	Jordan	17-JUN-86	jhenderson@gmail.com	0788654283	14-MAY-12
8	22	Stones	John	17-DEC-87	jstones.com	0721333019	14-MAY-12

BEGIN

INSERT INTO db_categories

VALUES (sec_cat.nextval,'Music');

INSERT INTO db_categories

VALUES (sec_cat.nextval,'Dance');

INSERT INTO db_categories

VALUES (sec_cat.nextval,'IT');

INSERT INTO db_categories

VALUES (sec_cat.nextval,'Politics');

INSERT INTO db_categories

VALUES (sec_cat.nextval,'Entertainment');

END;

/

SELECT *

FROM db_categories;

	⚡ CATEGORY_ID	⚡ CATEGORY_NAME
1	1	Music
2	2	Dance
3	3	IT
4	4	Politics
5	5	Entertainment
6	28	Categorie proba

BEGIN

INSERT INTO db_roles

VALUES (sec_rol.nextval,'Admin');

INSERT INTO db_roles

VALUES (sec_rol.nextval,'Editor');

INSERT INTO db_roles

VALUES (sec_rol.nextval,'User');

INSERT INTO db_roles

VALUES (sec_rol.nextval,'Requesting');

END;

/

SELECT *

FROM db_roles;

	ROLE_ID	ROLE_NAME
1	1	Admin
2	2	Editor
3	3	User
4	4	Requesting

BEGIN

INSERT INTO db_groups

VALUES (sec_gro.nextval,'I <3 BD','Grupul pasionatilor de BD.',TO_DATE('17-06-2014','dd-mm-yyyy'),3,1);

INSERT INTO db_groups

VALUES (sec_gro.nextval,'Muzica clasica','Grupul pentru iubitorii de muzica clasica.',TO_DATE('23-08-2014','dd-mm-yyyy'),1,2);

INSERT INTO db_groups

VALUES (sec_gro.nextval,'iUmor','Grup destinat glumelor.',TO_DATE('02-08-2014','dd-mm-yyyy'),5,3);

```
INSERT INTO db_groups
```

```
VALUES (sec_gro.nextval,'C++','Tips and Tricks for C++.',TO_DATE('22-05-2014','dd-mm-yyyy'),3,1);
```

```
INSERT INTO db_groups
```

```
VALUES (sec_gro.nextval,'Scoala de dans','Lectii de dans si nu numai!',TO_DATE('24-11-2015','dd-mm-yyyy'),2,4);
```

```
END;
```

```
/
```

```
SELECT *
```

```
FROM db_groups;
```

	GROUP_ID	GROUP_NAME	GROUP_DESCRIPTION	GROUP_START_DATE	CATEGORY_ID	OWNER_ID
1	1	I <3 BD	Grupul pasionatilor de BD.	17-JUN-14	3	1
2	2	Muzica clasica	Grupul pentru iubitorii de muzica clasica.	23-AUG-14	1	2
3	3	iUmor	Grup destinat qlumelor.	02-AUG-14	5	3
4	4	C++	Tips and Tricks for C++.	22-MAY-14	3	1
5	5	Scoala de dans	Lectii de dans si nu numai!	24-NOV-15	2	4
6	21	JAVA	Grupul pasionatilor de JAVA.	09-JAN-21	3	6

```
BEGIN
```

```
INSERT INTO db_groups_users
```

```
VALUES (1,1,1,TO_DATE('17-06-2014','dd-mm-yyyy'));
```

```
INSERT INTO db_groups_users
```

```
VALUES (1,4,1,TO_DATE('22-05-2014','dd-mm-yyyy'));
```

```
INSERT INTO db_groups_users  
VALUES (2,2,1,TO_DATE('23-08-2014','dd-mm-yyyy'));
```

```
INSERT INTO db_groups_users  
VALUES (3,3,1,TO_DATE('02-08-2014','dd-mm-yyyy'));
```

```
INSERT INTO db_groups_users  
VALUES (4,5,1,TO_DATE('24-11-2015','dd-mm-yyyy'));
```

```
INSERT INTO db_groups_users  
VALUES (5,1,3,TO_DATE('18-06-2014','dd-mm-yyyy'));
```

```
INSERT INTO db_groups_users  
VALUES (5,4,3,TO_DATE('24-05-2014','dd-mm-yyyy'));
```

```
INSERT INTO db_groups_users  
VALUES (6,2,3,TO_DATE('25-08-2014','dd-mm-yyyy'));
```

```
INSERT INTO db_groups_users  
VALUES (7,2,2,TO_DATE('25-08-2014','dd-mm-yyyy'));
```

```
INSERT INTO db_groups_users  
VALUES (7,3,3,TO_DATE('10-08-2014','dd-mm-yyyy'));
```

```

INSERT INTO db_groups_users
VALUES (7,5,3,TO_DATE('27-11-2015','dd-mm-yyyy'));

END;

/

SELECT *
FROM db_groups_users;

```

	USER_ID	GROUP_ID	ROLE_ID	JOIN_DATE_IN_GROUP
1	1	1	1	17-JUN-14
2	1	4	1	22-MAY-14
3	2	2	1	23-AUG-14
4	3	3	1	02-AUG-14
5	4	5	1	24-NOV-15
6	5	1	3	18-JUN-14
7	5	4	3	24-MAY-14
8	6	2	3	25-AUG-14
9	7	2	2	25-AUG-14
10	7	3	3	10-AUG-14
11	7	5	3	27-NOV-15

```

BEGIN

```

```

INSERT INTO db_activities
VALUES (sec_act.nextval,'Curs1','Introdúcere in C++',TO_DATE('01-07-2014','dd-mm-yyyy'),1,4);

```

```

INSERT INTO db_activities
VALUES (sec_act.nextval,'Curs1','Introdúcere in BD',TO_DATE('01-07-2014','dd-mm-yyyy'),1,1);

```

```
INSERT INTO db_activities
```

```
VALUES (sec_act.nextval,'Lectie de Samba','Invatam Samba!',TO_DATE('01-12-2015','dd-mm-yyyy'),4,5);
```

```
INSERT INTO db_activities
```

```
VALUES (sec_act.nextval,'Concurs de stand-up','Poti castiga 1000$!',TO_DATE('01-09-2014','dd-mm-yyyy'),3,3);
```

```
INSERT INTO db_activities
```

```
VALUES (sec_act.nextval,'Curs2','Struct in C++',TO_DATE('08-07-2014','dd-mm-yyyy'),1,4);
```

```
INSERT INTO db_activities
```

```
VALUES (sec_act.nextval,'Curs3','Class in C++',TO_DATE('15-07-2014','dd-mm-yyyy'),1,4);
```

```
INSERT INTO db_activities
```

```
VALUES (sec_act.nextval,'Curs2','BD Design',TO_DATE('01-07-2014','dd-mm-yyyy'),1,1);
```

```
INSERT INTO db_activities
```

```
VALUES (sec_act.nextval,'Q and A','Raspundem la intrebari',TO_DATE('02-07-2014','dd-mm-yyyy'),5,1);
```

```
INSERT INTO db_activities
```

```
VALUES (sec_act.nextval,'Q and A','Raspundem la intrebari',TO_DATE('09-07-2014','dd-mm-yyyy'),5,1);
```

```
INSERT INTO db_activities
```

```
VALUES (sec_act.nextval,'Q and A','Raspundem la intrebari',TO_DATE('09-07-2014','dd-mm-yyyy'),5,4);
```

```
END;
```

```
/
```

```
SELECT *
```

```
FROM db_activities;
```

	ACTIVITY_ID	ACTIVITY_NAME	ACTIVITY_DESCRIPTION	START_DATE	USER_ID	GROUP_ID
1	1	Curs1	Introducere in C++	01-JUL-14	1	4
2	2	Curs1	Introducere in BD	01-JUL-14	1	1
3	3	Lectie de Samba	Invatam Samba!	01-DEC-15	4	5
4	4	Concurs de stand-up	Poti castiga 1000\$!	01-SEP-14	3	3
5	5	Curs2	Struct in C++	08-JUL-14	1	4
6	6	Curs3	Class in C++	15-JUL-14	1	4
7	7	Curs2	BD Design	01-JUL-14	1	1
8	21	Q and A	Raspundem la intrebari	02-JUL-14	5	1
9	22	Q and A	Raspundem la intrebari	09-JUL-14	5	1
10	23	Q and A	Raspundem la intrebari	09-JUL-14	5	4

6)

```
-- 6
```

```
-- Afisati o lista cu varstele membrilor crescator unui grup al carui id este dat ca parametru,
```

```
-- afisand la final si media lor.
```

```
CREATE OR REPLACE FUNCTION functie_1
```

```
(groupId db_groups.group_id%TYPE)
```

```
RETURN NUMBER IS
```



```

medie NUMBER := 0;
TYPE vector IS VARRAY(20) OF NUMBER;
t vector:= vector();
BEGIN

SELECT ROUND((sysdate - u.birth_date)/365)
BULK COLLECT INTO t
FROM db_groups g, db_groups_users gu, db_users u
WHERE g.group_id = 1
AND u.user_id = gu.user_id
AND gu.group_id = g.group_id
ORDER BY 1;

FOR i IN t.FIRST..t.LAST LOOP
    DBMS_OUTPUT.PUT_LINE(i || ' ' || t(i));
    medie := medie + t(i);
END LOOP;

DBMS_OUTPUT.PUT_LINE('==');
medie := ROUND(medie/t.COUNT);

RETURN medie;
END functie_1;
/
-- Apelare

```

```
BEGIN
DBMS_OUTPUT.PUT_LINE('Media grupului cu id-ul 1 este ' || functie_1(1));
END;
/
```

PL/SQL procedure successfully completed.

1. 30

2. 48

===

Media grupului cu id-ul 1 este 39

7)

-- 7

-- Pentru fiecare grup afisati ziua din saptamana in care au s-au inscris in grup

-- cele mai multe persoane. Afisati lista cu numele persoanelor respective, precum
si varsta.

```
CREATE OR REPLACE PROCEDURE procedura_1 IS TYPE tablou IS TABLE OF
VARCHAR2(20);
```

```
v_day tablou;
```

```
v_nr NUMBER;
```

```
loc NUMBER;
```

```
vechime NUMBER;
```

```
BEGIN
```

```
FOR v_gro IN (SELECT group_id id, group_name name FROM db_groups) LOOP
```

```

SELECT COUNT(*)
INTO v_nr
FROM db_groups_users gu
WHERE gu.group_id = v_gro.id;

IF v_nr = 1 THEN
    DBMS_OUTPUT.PUT_LINE('-----');
    DBMS_OUTPUT.PUT_LINE('In grupul ' || v_gro.id || ' este doar owner-ul grupului!');
    DBMS_OUTPUT.PUT_LINE('-----');
ELSE

SELECT day
BULK COLLECT INTO v_day
FROM
(SELECT gu.user_id, TO_CHAR(gu.join_date_in_group, 'DAY') day
FROM db_groups_users gu
WHERE gu.group_id = v_gro.id)
GROUP BY day
HAVING COUNT(user_id) = (SELECT MAX(nr)
                        FROM (SELECT COUNT(user_id) nr
                              FROM (SELECT user_id, TO_CHAR(join_date_in_group, 'DAY')
                                day
                                FROM db_groups_users

```

```

        WHERE group_id = v_gro.id)
        GROUP BY day));

DBMS_OUTPUT.PUT_LINE('-----');
DBMS_OUTPUT.PUT_LINE(v_gro.id || ' ' || v_gro.name || ' ' || v_day(1));

loc := 1;
vechime := -1;

        FOR ang IN (SELECT u.user_id, TO_CHAR(gu.join_date_in_group, 'DAY') day,
u.last_name nume ,
        ROUND((sysdate-u.birth_date)/365) vechime
                FROM db_users u, db_groups_users gu
                WHERE group_id = v_gro.id
                AND u.user_id = gu.user_id
                ORDER BY gu.join_date_in_group) LOOP

        IF ang.day = v_day(1) THEN
        vechime := ang.vechime;

        DBMS_OUTPUT.PUT_LINE('nume: ' || ang.nume || ', varsta: ' ||
ROUND(ang.vechime) || ' de ani');

        END IF;

        END LOOP;

```

```
END IF;  
END LOOP;  
  
END procedura_1;  
/  
-- apelare  
BEGIN  
procedura_1();  
END;  
/
```

```

-----
1 I <3 BD TUESDAY
nume: Petersen, varsta: 48 de ani
-----
2 Muzica clasica MONDAY
nume: Henderson, varsta: 35 de ani
nume: Maxwell, varsta: 32 de ani
-----
3 iUmor SATURDAY
nume: Rodriguez, varsta: 35 de ani
-----
4 C++ THURSDAY
nume: Petersen, varsta: 48 de ani
-----
5 Scoala de dans TUESDAY
nume: Thompson, varsta: 43 de ani
-----
In grupul 21 este doar owner-ul grupului!
-----

```

8)

```

-- 8
-- Pentru un user al carui nume este dat ca parametru sa se afiseze
-- grupurile in care este inscris, categoria/categoriile dominanta/dominante.
-- Sa se returneze numarul de grupuri in care este inscris.
-- afisand la final si media lor.
CREATE OR REPLACE FUNCTION functie_2
(nume db_users.last_name%TYPE,
prenume db_users.first_name%TYPE)

```

```

RETURN db_categories.category_name%TYPE IS
TYPE vector IS VARRAY(20) OF db_groups.group_name%TYPE;
categId db_categories.category_id%TYPE;
categName db_categories.category_name%TYPE;
nr NUMBER;
t vector:= vector();
BEGIN

SELECT COUNT(u.user_id) INTO nr
FROM db_users u
WHERE UPPER(u.last_name) = UPPER(ume)
AND UPPER(u.first_name) = UPPER(prenume);

IF nr = 1 THEN

    SELECT g.group_name ume
    BULK COLLECT INTO t
    FROM db_users u, db_groups_users gu, db_groups g
    WHERE u.user_id = gu.user_id
    AND gu.group_id = g.group_id
    AND UPPER(u.last_name) = UPPER(ume)
    AND UPPER(u.first_name) = UPPER(prenume);
    nr := t.COUNT;

    IF t.COUNT = 0 THEN RAISE_APPLICATION_ERROR(-20000,'Userul nu este in
niciun grup');

```

```

ELSE
    DBMS_OUTPUT.PUT_LINE('===== GRUPURI =====');
    FOR i IN t.FIRST..t.LAST LOOP
        DBMS_OUTPUT.PUT_LINE(t(i));
    END LOOP;

SELECT c.category_name BULK COLLECT INTO t
FROM db_users u, db_groups_users gu, db_groups g, db_categories c
WHERE u.user_id = gu.user_id
AND gu.group_id = g.group_id
AND g.category_id = c.category_id
AND UPPER(u.last_name) = UPPER('Petersen')
AND UPPER(u.first_name) = UPPER('Jordan')
GROUP BY c.category_name
HAVING COUNT(g.category_id) = ( SELECT MAX(COUNT(g.category_id))
                                FROM db_users u, db_groups_users gu, db_groups g
                                WHERE u.user_id = gu.user_id
                                AND gu.group_id = g.group_id
                                AND UPPER(u.last_name) = UPPER('Petersen')
                                AND UPPER(u.first_name) = UPPER('Jordan')
                                GROUP BY g.category_id);
DBMS_OUTPUT.PUT_LINE('===== CATEGORII DOMINANTE =====');
FOR i IN t.FIRST..t.LAST LOOP
    DBMS_OUTPUT.PUT_LINE(t(i));

```



```

END LOOP;

DBMS_OUTPUT.PUT_LINE('=====');

RETURN nr;

END IF;

ELSE

    IF nr = 0 THEN RAISE_APPLICATION_ERROR(-20000,'Nu exista niciun user cu
acest nume!');

    ELSE

        RAISE_APPLICATION_ERROR(-20001,'Exista mai multi useri cu acest nume!');

    END IF;

END IF;

END functie_2;

/

-- apel1

BEGIN

DBMS_OUTPUT.PUT_LINE('Numarul de grupuri in care este userul este ' ||
functie_2('Petersen','Jordan'));

END;

/

```

```
===== GRUPURI =====
```

```
I <3 BD
```

```
C++
```

```
===== CATEGORII DOMINANTE =====
```

```
IT
```

```
=====
```

```
Numarul de grupuri in care este userul este 2
```

```
-- apel2
```

```
BEGIN
```

```
DBMS_OUTPUT.PUT_LINE(funcție_2('Peterswwwen','Jordan'));
```

```
END;
```

```
/
```

```
Error report -
```

```
ORA-20000: Nu exista niciun user cu acest nume!
```

```
ORA-06512: at "STEFAN.FUNCTIE_2", line 60
```

```
ORA-06512: at line 3
```

```
20000. 00000 - "%s"
```

```
*Cause:      The stored procedure 'raise_application_error'  
              was called which causes this error to be generated.
```

```
*Action:     Correct the problem as described in the error message or contact  
              the application administrator or DBA for more information.
```

```
-- apel3
```

```
BEGIN
```

```
DBMS_OUTPUT.PUT_LINE(funcție_2('Stones','John'));
```

```
END;
```

```
/
```

```
Error report -
ORA-20000: Userul nu este in niciun grup
ORA-06512: at "STEFAN.FUNCTIE_2", line 27
ORA-06512: at line 4
20000. 00000 - "%s"
*Cause:      The stored procedure 'raise_application_error'
              was called which causes this error to be generated.
*Action:     Correct the problem as described in the error message or contact
              the application administrator or DBA for more information.
```

```
-- apel4
```

```
BEGIN
```

```
DBMS_OUTPUT.PUT_LINE(funcție_2('Henderson','Jordan'));

```

```
END;
```

```
/
```

```
Error report -
ORA-20001: Exista mai multi useri cu acest nume!
ORA-06512: at "STEFAN.FUNCTIE_2", line 62
ORA-06512: at line 5
```

9)

```
-- 9
```

```
-- Pentru un grup al carui nume este dat ca parametru afisati lista celor mai activi
membrii,
```

```
-- precum si rolul lor in grupul respectiv.
```

```
CREATE OR REPLACE PROCEDURE procedura_2(v_nume
db_groups.group_name%TYPE) IS
```

```
idGroup db_groups.group_id%TYPE;
```

```

numeCateg db_categories.category_name%TYPE;

BEGIN

SELECT group_id, c.category_name INTO idGroup, numeCateg
FROM db_groups g, db_categories c
WHERE group_name = v_num
AND g.category_id = c.category_id;

    DBMS_OUTPUT.PUT_LINE('==== ' || idGroup || ' ' || v_num || ' ' ||
numeCateg || '====');

FOR c_usr IN
(
SELECT u.user_id id, u.last_name nume, u.first_name prenume, r.role_name rol
FROM db_users u, db_groups_users gu, db_groups g, db_roles r
WHERE u.user_id = gu.user_id
AND gu.group_id = g.group_id
AND gu.role_id = r.role_id
AND g.group_id = idGroup
AND u.user_id IN
(
SELECT a.user_id
FROM db_activities a
WHERE a.group_id = idGroup
GROUP BY a.group_id, a.user_id

```

```

HAVING COUNT(a.activity_id) = (SELECT MAX(COUNT(activity_id))
                                FROM db_activities
                                WHERE group_id = idGroup
                                GROUP BY group_id, user_id))) LOOP

```

```

    DBMS_OUTPUT.PUT_LINE(c_usr.id || '. ' || c_usr.numc || ' ' || c_usr.prenume
    || ' - ' || c_usr.rol);

```

```

END LOOP;

```

```

EXCEPTION

```

```

    WHEN NO_DATA_FOUND THEN

```

```

        RAISE_APPLICATION_ERROR(-20000,'Nu exista grup cu acest nume!');

```

```

    WHEN TOO_MANY_ROWS THEN

```

```

        RAISE_APPLICATION_ERROR(-20001,'Exista mai multe grupuri cu acest nume!');

```

```

    WHEN OTHERS THEN

```

```

        RAISE_APPLICATION_ERROR(-20002,'Alta eroare!');

```

```

END procedura_2;

```

```

/

```

```

-- apelare

```

```

BEGIN

```

```

    procedura_2('I <3 BD');

```

```
procedura_2('Grup care nu exista');
```

```
END;
```

```
/
```

```
===== 1 I <3 BD IT =====
```

```
1. Petersen Jordan - Admin
```

```
5. Wilson Jack - User
```

```
Error report -
```

```
ORA-20000: Nu exista grup cu acest nume!
```

```
ORA-06512: at "STEFAN.PROCEDURA_2", line 38
```

```
ORA-06512: at line 3
```

```
20000. 00000 - "%s"
```

```
*Cause:      The stored procedure 'raise_application_error'  
              was called which causes this error to be generated.
```

```
*Action:     Correct the problem as described in the error message or contact  
              the application administrator or DBA for more information.
```

10)

```
-- 10
```

```
-- Creati un trigger care sa nu permita utilizatorului STEFAN sa
```

```
-- stearga date din tabelul db_categories in acest an.
```

```
CREATE OR REPLACE TRIGGER trigger_1
```

```
BEFORE DELETE ON db_categories
```

```
BEGIN
```

```
IF UPPER(USER) = 'STEFAN' AND TO_CHAR(sysdate,'yyyy') = '2021' THEN
```

```
RAISE_APPLICATION_ERROR(-20000,'Nu aveti voie sa stergeti din acest tabel de  
pe acest utilizator in 2021!');
```

```
END IF;
```

```
END;
```

/

```
DELETE FROM db_categories
```

```
WHERE category_id = 28;
```

Error report -

ORA-20000: Nu aveti voie sa stergeti din acest tabel de pe acest utilizator sambata!

ORA-06512: at "STEFAN.TRIGGER_1", line 3

ORA-04088: error during execution of trigger 'STEFAN.TRIGGER_1'

11)

-- 11

-- Nu se poate insera in tabelul db_activities o activitate creata de un utilizator

-- care nu se afla in grupul respectiv.

```
CREATE OR REPLACE TRIGGER trigger_2
```

```
BEFORE INSERT OR UPDATE ON db_activities
```

```
FOR EACH ROW
```

```
BEGIN
```

```
IF functie_aux(:NEW.group_id,:NEW.user_id) = 0 THEN
```

```
RAISE_APPLICATION_ERROR(-20002,'Userul cu id-ul ' || :NEW.user_id || ' nu se  
afla in grupul cu id-ul ' ||
```

```
:NEW.group_id);
```

```
END IF;
```

```
END;
```

/

```

CREATE OR REPLACE FUNCTION functie_aux
(idGrup db_groups_users.group_id%TYPE,
idUser db_groups_users.user_id%TYPE)
RETURN NUMBER IS
nr NUMBER;
BEGIN
SELECT COUNT(1) INTO nr
FROM db_groups_users
WHERE group_id = idGrup
AND user_id = idUser;
RETURN nr;
END functie_aux;
/
-- proba
INSERT INTO db_activities
VALUES (sec_act.nextval,'Q and A','Raspundem la intrebari',TO_DATE('16-07-
2014','dd-mm-yyyy'),3,4);

```

Error report -

```

ORA-20002: Userul cu id-ul 3 nu se afla in grupul cu id-ul 4
ORA-06512: at "STEFAN.TRIGGER_2", line 3
ORA-04088: error during execution of trigger 'STEFAN.TRIGGER_2'

```

-- in plus

```

CREATE OR REPLACE TRIGGER trigger_3
AFTER INSERT ON db_groups

```



```

FOR EACH ROW
BEGIN
    INSERT INTO db_groups_users
    VALUES (:NEW.owner_id, :NEW.group_id, 1, :NEW.group_start_date);
END;
/
-- proba
INSERT INTO db_groups
VALUES (sec_gro.nextval,'JAVA','Grupul pasionatilor de
JAVA.',TO_DATE(sysdate,'dd-mm-yyyy'),3,6);

```

```

3      6      41      1 09-JAN-21

```

12)

```

-- 12
-- Salvati informatii despre erori in tabelul db_errors

```

```

CREATE TABLE db_errors (
    user_id VARCHAR(30),
    bd_name VARCHAR(30),
    error_name VARCHAR(300),
    error_date DATE DEFAULT SYSDATE
);

```

```

CREATE OR REPLACE TRIGGER trigger_errors

```

AFTER SERVERERROR ON SCHEMA

BEGIN

INSERT INTO db_errors

VALUES (SYS.LOGIN_USER, SYS.DATABASE_NAME,
DBMS_UTILITY.FORMAT_ERROR_STACK, sysdate);

END;

/

BEGIN

RAISE_APPLICATION_ERROR(-20002,'Eroare custom');

END;

/

SELECT *

FROM db_errors;

	USER_ID	BD_NAME	ERROR_NAME
1	STEFAN XE		ORA-20002: Eroare customORA-06512: at line 2