```
--Exercitii tema (de jos)
--1
create or replace trigger ex1_sca
   before delete on dep_sca
begin
   if user != 'SCOTT' then
      raise_application_error(-20001, 'Nu aveti permisiuni sa stergeti din acest tabel');
   end if;
end;
/
drop trigger ex1_sca;


--2
create or replace trigger ex2_sca
   before update of commission_pct on emp_sca
   for each row
   declare
      salariu emp_sca.salary%type;
begin
   select salary into salariu
   from emp_sca
   where employee_id = :OLD.employee_id;
   if :NEW.commission_pct * salariu > salariu * 0.5 then
      raise_application_error (-20001, 'Nu se poate mari comisionul');
   end if;
end;
/
drop trigger ex2_sca;


--3
--a
alter table info_dept_sca
add numar number(5);
insert into info_dept_sca (numar)
select count(employee_id)
from employees e join departments d on (e.department_id = d.department_id)
where id = d.department_id;



--b
create or replace trigger ex3_sca
after insert or update of id_dept on info_emp_sca
for each row
begin
   if inserting then
      update info_dept_sca
      set numar = numar + 1
      where :new.id_dept = id;
```

```
        else
            update info_dept_sca
            set numar = numar + 1
            where :new.id_dept = id;
            update info_dept_sca
            set numar = numar - 1
            where :old.id_dept = id;
        end if;
end ex3_sca;
/

--5
--a
create table emp_test_sca (
employee_id number(6) primary key,
last_name varchar(30),
first_name varchar(30),
department_id number(4));

create table dept_test_sca (
department_id number(4) primary key,
department_name varchar(30));

insert into emp_test_sca (employee_id, last_name, first_name, department_id)
select employee_id, last_name, first_name, department_id from emp_sca;

insert into dept_test_sca (department_id, department_name)
select department_id, department_name from departments;

--b
create or replace trigger trigg_ex5_sca
after delete or update of department_id on dept_test_sca
for each row
begin
    if deleting then
        delete from emp_test_sca
        where department_id = :old.department_id;
        dbms_output.put_line('Stergere cu succes');
    else
        update emp_test_sca
        set department_id = :new.department_id
        where department_id = :old.department_id;
        dbms_output.put_line('Actualizare cu succes');
    end if;
end trigg_ex5_sca;
/

select count(*)
```
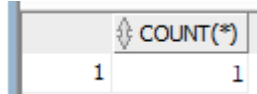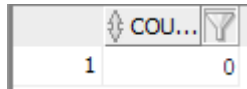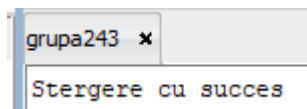
from emp_test_sca
where department_id = 40;

	COUNT(*)
1	1

delete from dept_test_sca
where department_id = 40;

	COU...
1	0

grupa243 ✕

Stergere cu succes

alter table emp_test_sca
add constraint dept_fk foreign key (department_id) references dept_test_sca(department_id);

--daca am conditie de foreign key atunci triggerul functioneaza bine
--daca se adauga si clauza on delete cascade atunci emp_test_sca devine mutating si triggerul nu functioneaza cum trebuie;
--am facut un compound trigger pentru a rezolva aceasta problema
create or replace trigger trigg_ex5_compound_sca
for delete or update of department_id on dept_test_sca
compound trigger
    type v_id is table of emp_test_sca.employee_id%type;
    v v_id;
before statement is
begin
    dbms_output.put_line('S-a declansat trigger-ul');
    select employee_id
    bulk collect into v
    from emp_test_sca
    where department_id = department_id;
end before statement;
after statement is
begin
    if deleting then
        for i in v.first..v.last loop
            delete from emp_test_sca
            where employee_id = v(i);
        end loop;
        dbms_output.put_line('Stergere cu succes');
    else
        for i in v.first..v.last loop
            update emp_test_sca
            set department_id = department_id
            where employee_id = v(i);

```
        end loop;
        dbms_output.put_line('Actualizare cu succes');
    end if;
end after statement;
end trigg_ex5_compound_sca;
/

--6
--a
create table error_sca (
user_id number(6),
nume_bd varchar2(25),
erori varchar2(30),
data date);

--b
create or replace trigger trigg_ex6_sca
    after SERVERERROR ON schema
declare
    n number;
    sql_text ora_name_list_t;
begin
    n := ora_sql_txt(sql_text);
    if n > 1000 then
        n := 1000;
    end if;
    insert into error_sca
    values (sys.login_user, sys.database_name, dbms_utility.format_error_stack, sysdate);
end trigg_ex6_sca;
/

--ex5 lab6
--g
create or replace trigger trig5_sca
    instead of insert or delete or update on v_info_sca
    for each row
begin
    if inserting then
        if :new.id is not null then
            insert into info_emp_sca
            values (:new.id, :new.nume, :new.prenume, :new.salariu, :new.id_dept);

            insert into info_dept_sca
            values (:new.id_dept, :new.nume_dep, :new.salariu, 1);
        end if;

        insert into info_dept_sca
        values (:new.id_dept, :new.nume_dep, 0, 0);
```

```
      elsif deleting then
         delete from info_emp_sca
         where id = :old.id;

         update info_dept_sca
         set plati = plati - :old.salariu
         where id = :old.id_dept;

      elsif updating ('salariu') then
         update info_emp_prof
         set salariu = :new.salariu
         where id = :old.id;

         update info_dept_sca
         set plati = plati - :old.salariu + :new.salariu
         where id = :old.id_dept;

      elsif updating ('id_dept') then
         update info_emp_sca
         set id_dept = :new.id_dept
         where id = :old.id;

         update info_dept_sca
         set plati = plati - :old.salariu
         where id = :old.id_dept;

         update info_dept_sca
         set plati = plati + :new.salariu
         where id = :new.id_dept;
      end if;
end trig5_sca;
/

--i
Numele angajatului nu se va modifica

--j
create or replace trigger trig5_sca
   instead of insert or delete or update on v_info_sca
   for each row
begin
   if inserting then
      if :new.id is not null then
         insert into info_emp_sca
         values (:new.id, :new.nume, :new.prenume, :new.salariu, :new.id_dept);

         insert into info_dept_sca
```

```
        values (:new.id_dept, :new.nume_dep, :new.salariu, 1);
    end if;

    insert into info_dept_sca
    values (:new.id_dept, :new.nume_dep, 0, 0);

elsif deleting then
    delete from info_emp_sca
    where id = :old.id;

    update info_dept_sca
    set plati = plati - :old.salariu
    where id = :old.id_dept;

elsif updating ('salariu') then
    update info_emp_prof
    set salariu = :new.salariu
    where id = :old.id;

    update info_dept_sca
    set plati = plati - :old.salariu + :new.salariu
    where id = :old.id_dept;

elsif updating ('id_dept') then
    update info_emp_sca
    set id_dept = :new.id_dept
    where id = :old.id;

    update info_dept_sca
    set plati = plati - :old.salariu
    where id = :old.id_dept;

    update info_dept_sca
    set plati = plati + :new.salariu
    where id = :new.id_dept;

elsif updating ('nume') then
    update info_emp_sca
    set nume = :new.nume
    where id = :new.id;

elsif updating ('prenume') then
    update info_emp_sca
    set prenume = :new.prenume
    where id = :new.id;

elsif updating ('nume_dep') then
    update info_dept_sca
```

```
        set nume_dep = :new.nume_dep
        where id = :new.id_dept;
    end if;
end trig5_sca;
/
```