### Ex 1

Create or replace function( role joaca.rol%type ) return NUMBER

Is

nr number := 0

dif date;

Cursor c is ( select id\_actor, data\_nastere from actor );

Aux number;

Begin

For actor in c loop

Select EXTRACT(YEAR FROM sysdate) - EXTRACT(YEAR FROM actor.data\_nastere) into dif from dual;

if dif >= 35 then

Select count(\*) into aux from joaca where cod\_actor = actor.id\_actor and rol = role;

If aux > 0 then

nr := nr + 1;

End if;

End if;

End loop;

return nr;

End;

/

### Ex 2

create or replace procedure (stagiune spectacol.stagiune%type)

is

cursor c is (select \* from spectacol where stagiune = stagiune);

cursor c\_actor is (select \* from);

pret number;

begin

for stag in c loop

dbms\_output.put\_line("- Nume spec " || stag.denumire);

select sum(pret) into pret from bilet\_vandut where cod\_spectacol = stag.id\_spectacol;

dbms\_output.put\_line("-- Total bilete vandute " || pret);

for actor in ( select \* from actor a join joaca j on a.id\_actor = j.cod\_actor join spectacol s on s.id\_spectacol = j.cod\_spectacol where s.id\_spectacol = stag.id\_spectacol ) loop

dbms\_output.put\_line("--- Actor: " || actor.nume);

end loop;

end loop;

end;

/

### Ex 3

create or replace trigger trg\_vandut

before insert on bilet\_vandut

declare

nr\_locuri sala.numar\_locuri%type;

nr\_locuri\_ocupate number;

begin

select numar\_locuri into nr\_locuri from sala s join spectacol sp on sp.cod\_sala = s.id\_sala;

select count(\*) into nr\_locuri\_ocupate from bilet\_vandut where cod\_spectacol = :new.cod\_spectacol;

if nr\_locuri\_ocupate >= nr\_locuri then

raise\_application\_error(-20001, 'Nu mai e locuri boss');

end if;

end;