

-Proiect-

-Sisteme de Gestiune a Bazelor de Date-

-Gestiunea unei firme IT-

Hadirca Dionisie

In cadrul acestui Proiect am ales sa modelez o baza de date a unei companii IT. Scopul gestiunii este de a simplifica modul de retinere a informatiei legate de lansarea unor produse inovative pe piata, angajatii care dezvoltă aceste produse, departamentele unde acestia lucreaza, echipamentul pe care il utilizeaza, proiectele si taskurile pe care le indeplinesc fiecare, precum si a clientilor care beneficiaza de aceste produse / solicita diverse servicii oferite de companie

In baza de date:

- Fiecare angajat poate avea doar un singur job
- Fiecare angajat poate lucra doar intr-un singur departament
- Mai multe sedii se pot afla intr-o locatie
- Departamente diferite nu pot lansa acelasi produs
- Este posibil ca nici un client sa nu beneficieze de un produs
- Fiecare angajat poate sa se implice doar intr-un singur proiect
- Mai multe departamente nu pot sa imparta acelasi echipament

Diagrama Entitate-Relatie

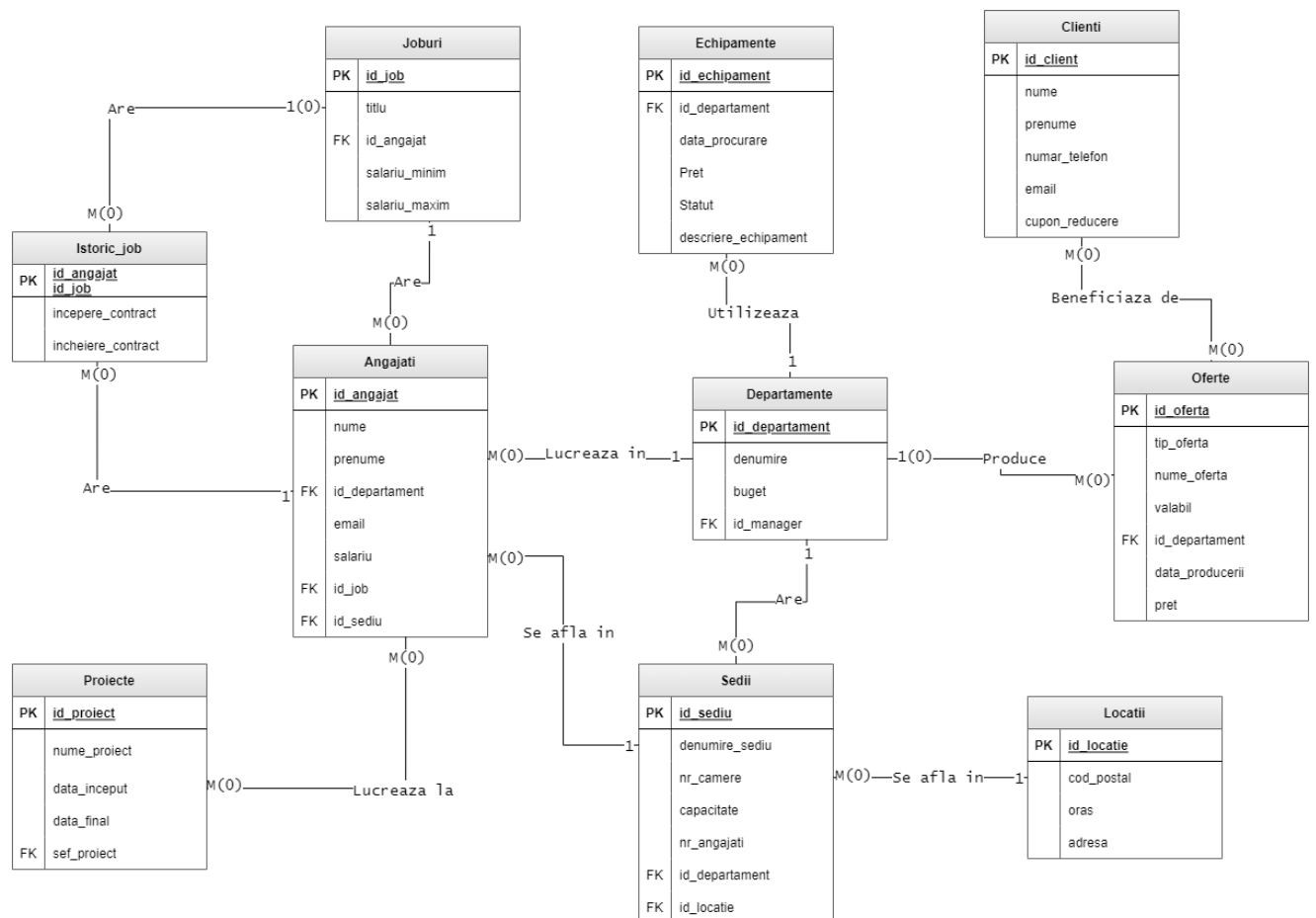
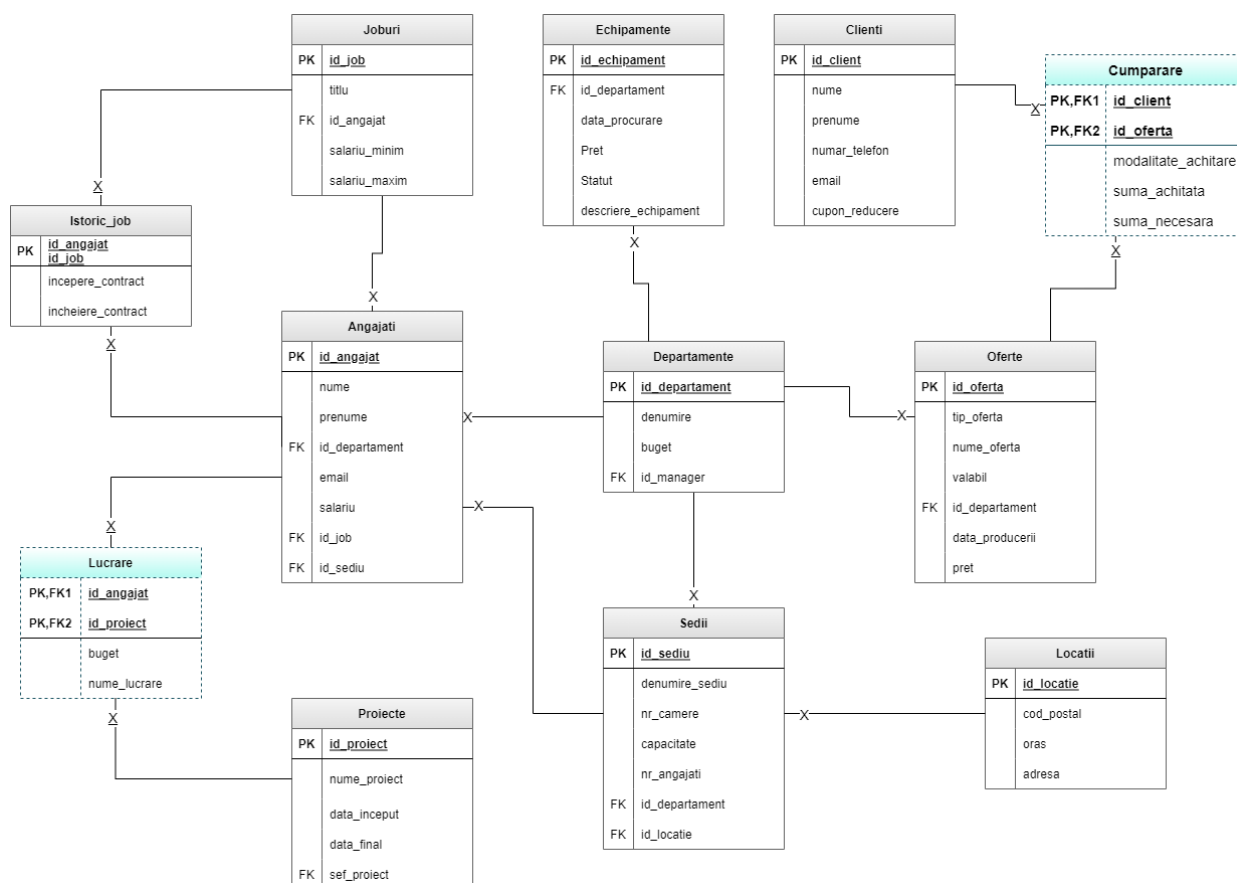


Diagrama Conceptuala



Creare Baza de date

--Dropare baza de date pentru debugging

```

drop table angajati cascade constraints;
drop table departamente cascade constraints;
drop table proiecte cascade constraints;
drop table joburi cascade constraints;
drop table istoric_job cascade constraints;
drop table echipamente cascade constraints;
drop table sedii cascade constraints;
drop table locatii cascade constraints;
drop table oferta cascade constraints;
drop table clienti cascade constraints;
drop table cumparare cascade constraints;
drop table lucrare cascade constraints;
drop table actualizari cascade constraints;

drop trigger modif_angajati;
drop trigger trigger_cupon;
drop sequence id_seq;
commit;

--Sfarsit Dropare

--Script crearea bazei de date Gestionarea unei firme IT
create table angajati(
id_angajat number(3),
nume varchar2(20) not null,
prenume varchar2(20) not null,
id_departament number(3) not null,
email varchar2(30),
salariu number(6),
id_job number(3),
id_sediu number(3),
constraint angajati_id_angajat_pk primary key(id_angajat)
);

create table departamente(
id_departament number(3),
denumire varchar2(30) not null,
buget number(10),
id_manager number(3) not null,
constraint departamente_id_departament_pk primary key (id_departament)
);

create table proiecte(
id_proiect number(3),
nume_proiect varchar2(30),
data_inceput date,
data_final date,
sef_proiect number(3) not null, --id-ul angajatului responsabil
constraint proiecte_id_proiect_pk primary key(id_proiect)
);

create table joburi(
id_job number(3),
titlu varchar2(50) not null,
salariu_minim number(6),
salariu_maxim number(6),
constraint joburi_id_job_pk primary key(id_job)
);

```

```

create table istoric_job(
id_job number(3) not null,
id_angajat number(3) not null,
incepere_contract date,
incheiere_contract date,
constraint istoric_job_pk primary key(id_angajat,id_job)
);

create table echipamente(
id_echipament number(3),
id_departament number(3) not null,
data_procurare date,
pret number(6),
statut varchar2(20),
descriere_echipament varchar2(40),
constraint echipament_id_echipament_pk primary key(id_echipament),
check (pret > 0)
);

create table sedii(
id_sediu number(3),
denumire_sediu varchar2(30),
nr_camere number(3),
capacitate number(4),
nr_angajati number(4),
id_departament number(3),
id_locatie number(3),
check (capacitate >= nr_angajati),
constraint sedii_id_sediu_pk primary key(id_sediu)
);

create table locatii(
id_locatie number(3),
cod_postal varchar2(10) not null,
oras varchar2(20),
adresa varchar2(50),
constraint locatii_id_locatie_pk primary key(id_locatie)
);

create table oferta(
id_oferta number(3),
tip_oferta varchar2(20),
nume_oferta varchar2(20) not null,
valabil number(3), --cate produse mai sunt valabile
id_departament number(3) not null,
data_producerii date not null,
pret number(10),
constraint oferta_id_oferta_pk primary key(id_oferta)
);

create table clienti(
id_client number(3),
nume varchar2(20) not null,
prenume varchar2(20) not null,
numar_telefon varchar2(20),
email varchar2(30),

```

```

cupon_reducere number(3),
constraint clienti_id_client_pk primary key(id_client)
);

create table cumparare(
id_client number(3) not null,
id_oferta number(3) not null,
constraint cumparare_pk primary key(id_client,id_oferta),
modalitate_achitare varchar2(20),
suma_achitata number(10),
suma_necesara number(10),
constraint cumparare_client_fk foreign key (id_client) references clienti (id_client),
constraint cumparare_oferta_fk foreign key (id_oferta) references oferta (id_oferta)
);

create table lucrare(
id_angajat number(3) not null,
id_proiect number(3) not null,
buget number(6),
nume_lucrare varchar2(50),
constraint lucrare_pk primary key(id_angajat,id_proiect),
constraint lucrare_id_angajat_fk foreign key(id_angajat) references
angajati(id_angajat),
constraint lucrare_id_proiect_fk foreign key(id_proiect) references proiecte(id_proiect)
);

create table actualizari(
    nume_user varchar2(30),
    data_actualizare date,
    descriere varchar2(100)
);

```

Inserare in Baza de Date

```

--Inserare date
create sequence id_seq
start with 1
increment by 1
minvalue 0
maxvalue 9999
nocycle;

--inserare departamente
insert into departamente
values
(id_seq.nextval, 'Marketing', 5000000, 2);

insert into departamente
values

```

```

(id_seq.nextval,'VR applications developement',7000000,1);

insert into departamente
values
(id_seq.nextval,'AR applications developement',6000000,3);

insert into departamente
values
(id_seq.nextval,'Web-Support',1000000,5);

insert into departamente
values
(id_seq.nextval,'Gadged creation',3000000,4);

insert into departamente
values
(id_seq.nextval,'3D Modelling',400000,7);

select * from departamente;

--inserare clienti
drop sequence id_seq;
create sequence id_seq
start with 1
increment by 1
minvalue 0
maxvalue 9999
nocycle;

insert into clienti
values
(id_seq.nextval,'Sparrow','Jack','0729285445','jacksparrow@s.unibuc.ro',20);

insert into clienti
values
(id_seq.nextval,'Vader','Dart','074546557','dartvader@s.unibuc.ro',0);

insert into clienti
values
(id_seq.nextval,'Chan','Jackie','071124589','jackiechan@s.unibuc.ro',0);

insert into clienti
values
(id_seq.nextval,'Stark','Tony','074488662','tonystark@gmail.com',0);

insert into clienti
values
(id_seq.nextval,'McFly','Marty','0769698448','martivirgula@gmail.com',0);

select * from clienti;

--inserare joburi
drop sequence id_seq;
create sequence id_seq
start with 1
increment by 1

```

```
minvalue 0
maxvalue 9999
nocycle;

insert into joburi
values
(id_seq.nextval, 'Full Stack Developer', 2000, 2000);

insert into joburi
values
(id_seq.nextval, 'C++ Programmer', 6000, 6000);

insert into joburi
values
(id_seq.nextval, 'Salesman', 2500, 2500);

insert into joburi
values
(id_seq.nextval, 'Application Tester', 3000, 3000);

insert into joburi
values
(id_seq.nextval, 'Robotics Engineer', 5000, 5000);

insert into joburi
values
(id_seq.nextval, 'Graphic designer', 4500, 4500);

insert into joburi
values
(id_seq.nextval, '3D designer', 4300, 4300);

select * from joburi;

--inserare locatii
drop sequence id_seq;
create sequence id_seq
start with 1
increment by 1
minvalue 0
maxvalue 9999
nocycle;

insert into locatii
values
(id_seq.nextval, '620129', 'Singerei', 'str. Independentei 15');

insert into locatii
values
(id_seq.nextval, '621332', 'Balti', 'str. Bucuriei 2');

insert into locatii
values
(id_seq.nextval, '410192', 'Chisinau', 'str Bunei-Dispozitii 18');

insert into locatii
```



```

values
(id_seq.nextval, '123456', 'Chisinau', 'str. Intelectului 20');

insert into locatii
values
(id_seq.nextval, '653122', 'Balti', 'str. Stefan-cel-Mare 1');

select * from locatii;

--inserare oferta
drop sequence id_seq;
create sequence id_seq
start with 1
increment by 1
minvalue 0
maxvalue 9999
nocycle;

insert into oferta
values
(id_seq.nextval, 'produs', 'Halloween game', 15, 2, to_date('21/05/20', 'DD/MM/YY'), 10000);

insert into oferta
values
(id_seq.nextval, 'serviciu', 'Bonus per
Cumparare', 300, 1, to_date('10/10/19', 'DD/MM/YY'), 1000);

insert into oferta
values
(id_seq.nextval, 'produs', 'Robotic Costume', 0, 5, to_date('31/12/19', 'DD/MM/YY'), 2000000);

insert into oferta
values
(id_seq.nextval, 'serviciu', 'Gestiunea unui
site', 3, 4, to_date('21/05/20', 'DD/MM/YY'), 300);

insert into oferta
values
(id_seq.nextval, 'produs', 'IT-Kinder Surprise', 999, 3, to_date('11/07/20', 'DD/MM/YY'), 20);

insert into oferta
values
(id_seq.nextval, 'produs', 'Pacman VR', 999, 2, to_date('17/05/17', 'DD/MM/YY'), 5000);

select * from oferta;

--inserare cumparare
drop sequence id_seq;
create sequence id_seq
start with 1
increment by 1
minvalue 0
maxvalue 9999
nocycle;

insert into cumparare
values

```

```

(1,2,'in rate',800,1100);

insert into cumparare
values
(1,4,'direct',300,300);

insert into cumparare
values
(2,5,'direct',20,(select pret from oferta where id_oferta = 5));

insert into cumparare
values
(5,1,'in rate',1000,11000);

insert into cumparare
values
(3,2,'in rate',300,1100);

insert into cumparare
values
(1,5,'direct',20,(select pret from oferta where id_oferta = 5));

insert into cumparare
values
(4,4,'direct',300,300);

insert into cumparare
values
(2,3,'direct',2000000,2000000);

insert into cumparare
values
(5,4,'direct',300,300);

insert into cumparare
values
(1,1,'in rate',5500,11000);

select * from cumparare;

--inserare echipamente
drop sequence id_seq;
create sequence id_seq
start with 1
increment by 1
minvalue 0
maxvalue 9999
nocycle;

insert into echipamente
values
(id_seq.nextval,1,to_date('10/03/15','DD/MM/YY'),50000,'vechi','set ochelari VR 10
perechi');

insert into echipamente
values

```

```

(id_seq.nextval,4,to_date('11/11/17','DD/MM/YY'),20000,'mediu-nou','set circuite pe baza
arduino');

insert into echipamente
values
(id_seq.nextval,3,to_date('10/06/18','DD/MM/YY'),200000,'mediu-nou','set server
holder');

insert into echipamente
values
(id_seq.nextval,2,to_date('05/01/19','DD/MM/YY'),40000, 'nou','pachet tehnologic pentru
AR');

insert into echipamente
values
(id_seq.nextval,4,to_date('20/05/18','DD/MM/YY'),100000,'mediu-nou','set buisness PC');

insert into echipamente
values
(id_seq.nextval,2,to_date('27/02/19','DD/MM/YY'),50000,'nou','ochelari AR, ultima
generatie');

select * from echipamente;

--inserare angajati
drop sequence id_seq;
create sequence id_seq
start with 1
increment by 1
minvalue 0
maxvalue 9999
nocycle;

insert into angajati
values
(id_seq.nextval,'Johnny','Depp',2,'johnnydepp@gmail.ro',
(select salariu_maxim from joburi where id_job = 2),2,1);

insert into angajati
values
(id_seq.nextval,'Dave','Prowse',1,'daveprorowse@gmail.com',
(select salariu_maxim from joburi where id_job = 3),3,3);

insert into angajati
values
(id_seq.nextval,'Jackie','Chan',3,'jackiechan@gmail.ro',
(select salariu_maxim from joburi where id_job = 4),4,4);

insert into angajati
values
(id_seq.nextval,'Robert','Downey',5,'robertdowneyjr@gmail.ro',
(select salariu_maxim from joburi where id_job = 5),5,5);

insert into angajati
values
(id_seq.nextval,'Scarlet','Johansson',4,'scarletkek@gmail.ro',

```

```

(select salariu_maxim from joburi where id_job = 1),1,2);

insert into angajati
values
(id_seq.nextval,'John','Cena',4,'johnцена@gmail.ro',4500,6,2);

insert into angajati
values
(id_seq.nextval,'Tom','Holland',6,'tomholland@gmail.ro',3000,7,6);

select * from angajati;

--inserare istoric job
insert into istoric_job
values
(1,5,to_date('18/02/15','DD/MM/YY'),null);

insert into istoric_job
values
(2,1,to_date('13/11/15','DD/MM/YY'),null);

insert into istoric_job
values
(3,2,to_date('29/12/16','DD/MM/YY'),null);

insert into istoric_job
values
(4,3,to_date('22/03/15','DD/MM/YY'),null);

insert into istoric_job
values
(5,4,to_date('11/11/16','DD/MM/YY'),null);

insert into istoric_job
values
(6,6,to_date('10/05/16','DD/MM/YY'),null);

select * from istoric_job;

--inserare sedii

drop sequence id_seq;
create sequence id_seq
start with 1
increment by 1
minvalue 0
maxvalue 9999
nocycle;

insert into sedii
values
(id_seq.nextval,'Sediu 1 App dev',10,10,1,2,1);

insert into sedii
values

```

```

(id_seq.nextval, 'Sediu 1 Web Supp', 5, 15, 2, 4, 1);

insert into sedii
values
(id_seq.nextval, 'Sediu 1 Sales', 7, 15, 1, 1, 1);

insert into sedii
values
(id_seq.nextval, 'Sediu 2 App dev', 10, 10, 1, 3, 2);

insert into sedii
values
(id_seq.nextval, 'Sediu 1 Robot. construction', 10, 20, 1, 5, 2);

insert into sedii
values
(id_seq.nextval, 'Sediu 1 3D Modelling', 1, 3, 1, 6, 3);

select * from sedii;

--inserare proiecte
drop sequence id_seq;
create sequence id_seq
start with 1
increment by 1
minvalue 0
maxvalue 9999
nocycle;

insert into proiecte
values
(id_seq.nextval, 'Halloween VR anturaj', to_date('01/09/20', 'DD/MM/YY'),
to_date('30/10/20', 'DD/MM/YY'), 1);

insert into proiecte
values
(id_seq.nextval, 'Robot Costume', to_date('03/05/18', 'DD/MM/YY'),
to_date('13/08/20', 'DD/MM/YY'), 4);

insert into proiecte
values
(id_seq.nextval, 'VR website', to_date('13/01/15', 'DD/MM/YY'),
to_date('20/02/15', 'DD/MM/YY'), 5);

insert into proiecte
values
(id_seq.nextval, 'Chemistry Lab VR', to_date('22/03/17', 'DD/MM/YY'),
to_date('17/08/19', 'DD/MM/YY'), 1);

insert into proiecte
values
(id_seq.nextval, 'Pacman VR', to_date('15/03/17', 'DD/MM/YY'),
to_date('16/03/17', 'DD/MM/YY'), 1);

select * from proiecte;

--inserare lucrare
insert into lucrare

```

```
values
(1,1,1000,'Logica programului in C#');

insert into lucrare
values
(6,1,3000,'Anturaj Unity');

insert into lucrare
values
(4,2,100000,'Detalii Corp Robot');

insert into lucrare
values
(1,2,3000,'MIPS ASSEMBLY logica');

insert into lucrare
values
(5,3,3000,'Front-End si Back-End');

insert into lucrare
values
(6,3,1500,'React, TS interactivitatea web-siteului');

insert into lucrare
values
(1,4,2500,'Logica Programului C++/C#');

insert into lucrare
values
(6,4,3000,'Model, Unreal Engine');

insert into lucrare
values
(1,5,100,'Logica Programului in Java');

insert into lucrare
values
(2,5,1,'Nu a facut nimic');

select * from lucrare;

commit;

alter table angajati
add constraint angajati_id_departament_fk foreign key (id_departament) references
departamente(id_departament);

alter table angajati
add constraint angajati_id_job_fk foreign key (id_job) references joburi(id_job);

alter table sedii
add constraint sedii_id_departament_fk foreign key (id_departament) references
departamente(id_departament);

alter table sedii
add constraint sedii_id_locatie_fk foreign key (id_locatie) references
locatii(id_locatie);
```

```

alter table oferta
add constraint oferta_id_departament_fk foreign key (id_departament) references
departamente(id_departament);

alter table echipamente
add constraint echipamente_id_departament_fk foreign key(id_departament) references
departamente(id_departament);

alter table istoric_job
add constraint istoric_job_id_angajat_fk foreign key (id_angajat) references
angajati(id_angajat);

alter table istoric_job
add constraint istoric_job_id_job_fk foreign key (id_job) references joburi(id_job);

alter table angajati
add constraint angajati_id_sediu_fk foreign key (id_sediu) references sedii(id_sediu);

--Sfarsit Script creare bazei de date Gestiunea unei firme IT

```

Cerintele exercitiilor 6 – 9

6.

Sa se obtina pentru fiecare oras toti angajatii (numele si prenumele) care au/au avut cel mult 2 inregistrari in tabelul istoric_job.

Pentru acesti angajati se vor afisa pe ecran proiectele la care au luat parte si departamentele in care activeaza

Sa se determine si sa se afiseze cel mai bun angajat care a lucrat la cele mai multe proiecte per fiecare oras. Daca sunt mai multi angajati care respecta acest criteriu, se vor afisa toti. (Se stie ca sunt mai putini de 10000 angajati).

7.

Pentru fiecare client se va obtine lista cumparaturilor si se va afisa pe ecran denumirea produselor achizitionate, numarul de produse si suma de bani cheltuita.

Apoi se va afisa suma totala de bani primita de companie pe toate produsele(ofertele) care s-au vandut pana la acel moment (daca o oferta se procura in rate, si nu a fost achitata pana la final, se va afisa suma_achitata pentru acea oferta.

Deasemenea se va tine cont de cate oferte a vandut fiecare departament si se va afisa departamentul cu cel mai mare nr de oferte vandute

Daca sunt mai multe departamente care respecta acest criteriu, se vor afisa toate.

8.

Pentru un angajat dat ca parametru se va apela o functie care returneaza descrierea echipamentului care este utilizat in departamentu in care activeaza angajatul respectiv.

Cand sunt mai multe echipamente intr-un departament se va crea o exceptie(raise)

9.

Pe ecran se vor afisa clientii si suma totala achitata pe produsele companiei, produsele respective fiind lansate de departamentele unde activeaza angajatii unui job dat ca parametru.

Pentru acel job se va afisa si angajatul care are cel mai mare salariu.

Toti angajatii s-au inteles intre ei si vor sa faca un cadou de Craciun acestor clienti si deci fiecare client va primi un cupon de reducere pentru urmatoarele potentiale cumparaturi in dependenta de cat de mult a achitat fiecare

Clientul care a achitat cel mai mult va primi un cupon de 20% reducere la orice oferta iar al i-lea client va primi un cupon pe baza raportului intre suma achitata de primul client si suma achitata de al i-lea client

Se va face update pe coloane

Pachet Cerinte 6,7,8,9

```
set serveroutput on;
create or replace package pachet_proiect is

    --Cerinta 6
    --Sa se obtina pentru fiecare oras toti angajatii (numele si prenumele) care au/au
    avut
    --cel mult 2 inregistrari in tabelul job-history.
    --Pentru acesti angajati se vor afisa pe ecran proiectele la care au luat parte
    --si departamentele in care activeaza
    --sa se determine si sa se afiseze cel mai bun angajat
    --care a lucrat la cele mai multe proiecte, per fiecare oras
    --daca sunt mai multi angajati care respecta acest criteriu, se vor afisa toti
    --(se stie ca sunt mai putini de 10000 angajati)
    procedure afis_best_angajati_per_oras;

    --Cerinta 7
    --Pentru fiecare client se va obtine lista cumparaturilor si se va afisa
    --pe ecran denumirea produselor achizitionate numarul de produse si suma de bani
    cheltuita
```



```

--Apoi se va afisa suma totala de bani primita de companie pe toate
proudsele(ofertele)
--care s-au vandut pana la acel moment (daca o oferta se procura in rate,
--si nu a fost achitata pana la final,se va afisa suma_achitata pentru acea oferta
--Deasemenea se va tine cont de cate oferte a vandut fiecare departament
--si se va afisa departamentul cu cel mai mare nr de oferte vandute
--Daca sunt mai multe departamente ce au acelasi nr si e maxim de oferte lansate,
se vor afisa toate
procedure clienti_oferte_departamente;

--Cerinta 8
--pentru un angajat dat ca parametru se va apela o functie care returneaza
--descrierea echipamentului care este utilizat in departamentul in care
--activeaza angajatul respectiv
--cand sunt mai multe echipamente intr-un departament se va crea o exceptie (raise)
function get_echipament(f_nume angajati.nume%type) return
echipamente.descriere_echipament%type;

--Cerinta 9
--Pe ecran se vor afisa clienti si suma totala achitata pe produsele companiei
--produsele respective fiind lansate de departamentele unde activeaza angajatii
unui job dat ca parametru
--pentru acel job se va afisa si angajatul care are cel mai mare salariu
--Toti angajatii s-au inteles intre ei si vor sa faca un cadou de craciun acestor
clienti, si deci
--fiecare client va primi un cupon de reducere pentru urmatoarele potentiale
cumparaturi
--in dependenta de cat de mult a achitat fiecare
--clientul care a achitat cel mai mult va primi un cupon de 20% reducere la orice
oferta
--iar al i-lea client va primi un cupon pe baza raportului intre suma achitata de
primul client si suma achitata de al i-lea client
--(20 / (suma[primul] / suma[i])). se va face update pe coloane
procedure update_cupoane_reducere(f_titlu joburi.titlu%type);
end;
/

create or replace package body pachet_proiect is

procedure afis_best_angajati_per_oras
is
    type tablou_orase is table of locatii.oras%type;
    type ang_record is record (r_id angajati.id_angajat%type,r_nume
angajati.nume%type, r_prenume angajati.prenume%type);
    type vector_angajati is varray(10000) of ang_record;

    t_orase tablou_orase := tablou_orase();
    t_angajati vector_angajati := vector_angajati();

    v_departament departamente.denumire%type;
    numar_angajati number;

    id_best_ang number;
    part_num number;

    nr_proiecte number:=0;
    idx number;
begin

```

```

select distinct oras bulk collect into t_orase from locatii;

select count(*) into numar_angajati from ANGAJATI;

for i in 1..numar_angajati loop
    t_angajati.extend;
end loop;

for i in t_orase.first..t_orase.last loop

    nr_proiecte := 0;

    dbms_output.put_line('-----');
    dbms_output.put_line('Orasul' || ' ' || t_orase(i) || ':');
    idx := 0;
    for j in (select id_angajat, nume, prenume
               from angajati a, departamente d, sedii s, locatii l
               where a.id_departament = d.id_departament
                   and d.id_departament = s.id_departament
                   and s.id_locatie = l.id_locatie
                   and l.oras = t_orase(i)
                   and (select count(*) from istoric_job
                       where id_angajat = a.id_angajat) <=2) loop

        idx := idx+1;
        t_angajati(idx).r_id := j.id_angajat;
        t_angajati(idx).r_nume := j.nume;
        t_angajati(idx).r_prenume := j.prenume;

    end loop;

    if idx = 0 then

        dbms_output.put_line('In acest oras nu lucreaza nimeni ');
        continue;
    end if;

    for j in 1..idx loop

        part_num:=0;
        dbms_output.new_line;

        select denumire into v_departament
        from departamente d, angajati a
        where a.id_angajat = t_angajati(j).r_id
        and d.id_departament = a.id_departament;

        dbms_output.put_line('Angajatul ' || t_angajati(j).r_nume || ' ' ||
t_angajati(j).r_prenume || ' lucreaza in departamentul ' || v_departament);
        dbms_output.put_line('Proiecte: ');

        for proiect in (select nume_proiect
                        from proiecte p, lucrare l
                        where p.id_proiect = l.id_proiect
                        and l.id_angajat = t_angajati(j).r_id) loop

            part_num := part_num + 1;

```

```

        dbms_output.put_line('>>>>' || proiect.nume_proiect);
    end loop;

    if part_num >= nr_proiecte then
        nr_proiecte := part_num;
    end if;

end loop;

if nr_proiecte <> 0 then
    dbms_output.new_line;
    dbms_output.put_line('Cei mai bun angajati in orasul ' || t_orase(i) ||
' cu un numar de ' || nr_proiecte || ' proiecte' || ':');
    for j in 1..idx loop

        select count(*) into part_num
        from angajati a, proiecte p, lucrare l
        where p.id_proiect = l.id_proiect
        and l.id_angajat = a.id_angajat
        and a.id_angajat = t_angajati(j).r_id;

        if part_num = nr_proiecte then
            dbms_output.put_line(t_angajati(j).r_nume || ' ' ||
t_angajati(j).r_prenume);
        end if;
    end loop;
end if;
end loop;
end;

procedure clienti_oferte_departamente
is

    cursor comparaturi (id_cl clienti.id_client%type)
    is
        select c.id_oferta, c.suma_achitata,c.suma_necesara
        from comparare c
        where c.id_client = id_cl;

    type tablou_clienti is table of clienti.id_client%type;
    type freq_oferte is varray(100) of number;

    t_freq_oferte freq_oferte := freq_oferte();
    t_clienti tablou_clienti := tablou_clienti();

    v_id_oferta comparare.id_oferta%type;
    v_suma_achitata comparare.suma_achitata%type;
    v_suma_necesara comparare.suma_necesara%type;
    v_id_departament departamente.id_departament%type;

    v_nume_oferta oferta.nume_oferta%type;
    v_nume_client clienti.nume%type;
    v_prenume_client clienti.prenume%type;
    v_nume_departament departamente.denumire%type;

    v_nr_oferte_client number;
    v_suma_per_client number;
    v_suma_totala number := 0;

```

```

v_max_oferte number:=0;

v_nr_departamente number :=0;

begin

select id_client bulk collect into t_clienti from clienti;
select count(*) into v_nr_departamente from departamente;
t_freq_oferte.extend(v_nr_departamente);
for i in 1..v_nr_departamente loop
    t_freq_oferte(i) := 0;
end loop;

for i in t_clienti.first..t_clienti.last loop
    dbms_output.put_line('-----');

    select nume,prenume into v_nume_client,v_prenume_client from clienti
where id_client = t_clienti(i);
    v_suma_per_client := 0;
    v_nr_oferte_client := 0;

    dbms_output.put_line(v_nume_client||' '||v_prenume_client);

    open comparaturi(t_clienti(i));

        loop
            fetch comparaturi into v_id_oferta, v_suma_achitata,
v_suma_necesara;
            exit when comparaturi%notfound;

            select nume_oferta into v_nume_oferta
            from oferta
            where id_oferta = v_id_oferta;

            select id_departament into v_id_departament
            from oferta
            where id_oferta = v_id_oferta;

            dbms_output.put_line('>>>>' || v_nume_oferta);

            v_suma_per_client := v_suma_per_client + v_suma_achitata;
            v_nr_oferte_client := v_nr_oferte_client + 1;
            t_freq_oferte(v_id_departament):=
t_freq_oferte(v_id_departament) + 1;

        end loop;
        dbms_output.put_line('Suma pe care a platit-o clientul respectiv : '
||v_suma_per_client);
        dbms_output.put_line('Numarul de produse achizitionate de acest
client :'||v_nr_oferte_client);
        v_suma_totala := v_suma_totala + v_suma_per_client;

    close comparaturi;
end loop;

for i in 1..v_nr_departamente loop
    if v_max_oferte < t_freq_oferte(i) then

```

```

        v_max_oferte := t_freq_oferte(i);
        v_id_departament := i;
    end if;
end loop;

select denumire into v_num_departament from departamente where
id_departament = v_id_departament;

dbms_output.put_line('-----');
dbms_output.put_line('Suma totala obtinuta in urma vanzarii tuturor
produselor :'||v_suma_totala);
dbms_output.put_line('Departamentul '||v_num_departament||' au vandut cele
mai multe oferte in numar de ' || v_max_oferte);
end;

function get echipament(f_num angajati. nume%type) return
echipamente.descriere_echipament%type
is
    v_desc_echipament echipamente.descriere_echipament%type;
    multe_echipamente exception;
    nr_echipamente number;
    v_prenume angajati. nume%type;

begin

    select count(*) into nr_echipamente
    from echipamente e, departamente d, angajati a
    where e.id_departament = d.id_departament
    and d.id_departament = a.id_departament
    and a.nume = f_num;

    if nr_echipamente > 1 then
        raise multe_echipamente;
    end if;

    --linia respectiva a fost adaugata doar pentru a evidentia exceptia too
many rows
    select prenume into v_prenume from angajati where nume = f_num;

    select e.descriere_echipament into v_desc_echipament
    from echipamente e, departamente d, angajati a
    where e.id_departament = d.id_departament
    and d.id_departament = a.id_departament
    and a.nume = f_num;

    dbms_output.put_line(v_prenume);
    return v_desc_echipament;

exception
    when multe_echipamente then
        dbms_output.put_line('Mai multe echipamente');
        return 'A fost tratata exceptia definita de mine
[multe_echipamente]';
    when no_data_found then
        dbms_output.put_line('Nu exista angajatul cu numele respectiv');
        return 'A fost tratata exceptia [no_data_found]';
    when too_many_rows then

```

```

        dbms_output.put_line('Sunt mai multi angajati cu numele
respectiv');
        return 'A fost tratata exceptia [too_many_rows]';
    end;

    procedure update_cupoane_reducere(f_titlu joburi.titlu%type)
    is
        type cl_record is record (r_cl_id clienti.id_client%type, r_cl_num
        clienti.num%type,r_cl_prenume clienti.prenume%type,r_total number);
        type tablou_clienti is table of cl_record;

        t_clienti tablou_clienti := tablou_clienti();
        v_red_max number:=20;
        v_raport number;
        v_num_ang angajati.num%type;
        v_prenume_ang angajati.prenume%type;
    begin

        select c.id_client,c.num,c.prenume,sum(suma_achitata) as total
        bulk collect into t_clienti
        from clienti c, cumparare cu,oferta o, departamente d, angajati a, joburi j
        where c.id_client = cu.id_client
        and cu.id_oferta = o.id_oferta
        and o.id_departament = d.id_departament
        and d.id_departament = a.id_departament
        and a.id_job = j.id_job
        and j.titlu = f_titlu
        group by c.id_client,c.num,c.prenume
        order by total desc;

        select a.num,a.prenume into v_num_ang,v_prenume_ang
        from angajati a, joburi j
        where a.id_job = j.id_job
        and j.titlu = f_titlu;

        dbms_output.put_line('Cel mai platit angajat la jobul '||f_titlu||' este '
||v_num_ang||' '||v_prenume_ang);
        dbms_output.put_line('-----');

        for i in t_clienti.first..t_clienti.last loop
            dbms_output.put_line('Clientul'||' '||t_clienti(i).r_cl_prenume|| '
'||t_clienti(i).r_cl_num || ' '||'are un total de '||t_clienti(i).r_total);
        end loop;

        update clienti
        set cupon_reducere = 20
        where id_client = t_clienti(1).r_cl_id;

        for i in t_clienti.first..t_clienti.last loop
            v_raport := t_clienti(1).r_total / t_clienti(i).r_total;
            v_raport := 20 / v_raport;

            update clienti
            set cupon_reducere = v_raport
            where id_client = t_clienti(i).r_cl_id;
        end loop;
    
```

```

exception
    when no_data_found then
        dbms_output.put_line('A fost tratata exceptia [no_data_found]');
        dbms_output.put_line('Nu exista astfel de job');
    when too_many_rows then
        dbms_output.put_line('A fost tratata exceptia [too_many_rows]');
        dbms_output.put_line('Sunt mai multi angajati cu salariu maxim');
        for i in t_clienti.first..t_clienti.last loop
            dbms_output.put_line('Clientul'||' '||t_clienti(i).r_cl_prenume|| '
'||t_clienti(i).r_cl_nume || ' '||'are un total de '||t_clienti(i).r_total);
        end loop;

        update clienti
        set cupon_reducere = 20
        where id_client = t_clienti(1).r_cl_id;

        for i in t_clienti.first..t_clienti.last loop
            v_raport := t_clienti(1).r_total / t_clienti(i).r_total;
            v_raport := 20 / v_raport;

            update clienti
            set cupon_reducere = v_raport
            where id_client = t_clienti(i).r_cl_id;
        end loop;
    when value_error then
        dbms_output.put_line('A fost tratata exceptia [value_error]');
        dbms_output.put_line('Nici un client nu a cumparat produse la care au
lucrat angajatii job-ului respectiv');
    end;

end;
/

--8
declare
    v_desc echipamente.descriere_echipament%type;
begin
    v_desc := pachet_proiect.get_echipament('Johnny');
    dbms_output.put_line(v_desc);
end;
/

declare
    v_desc echipamente.descriere_echipament%type;
begin
    v_desc := pachet_proiect.get_echipament('Denis');
    dbms_output.put_line(v_desc);
end;
/

insert into angajati values (10,'Tom','Prowse',1,'daveprorowse@gmail.com',
(select salariu_maxim from joburi where id_job = 3),3,3);

declare
    v_desc echipamente.descriere_echipament%type;
begin
    v_desc := pachet_proiect.get_echipament('Tom');
    dbms_output.put_line(v_desc);

```

```

end;
/

delete from angajati where id_angajat = 10;

--9
begin
    dbms_output.put_line('caz 1');
    pachet_proiect.update_cupoane_reducere('Full Stack Developer');
end;
/

begin
    dbms_output.put_line('caz 2');
    pachet_proiect.update_cupoane_reducere('3D designer');
end;
/

begin
    dbms_output.put_line('caz 3');

    pachet_proiect.update_cupoane_reducere('un job');
end;
/

insert into angajati values (10, 'temp', 'test', 2, 'test@', 6000, 2, 1);
begin
    dbms_output.put_line('caz 4');
    pachet_proiect.update_cupoane_reducere('C++ Programmer');
end;
/
delete from angajati where id_angajat = 10;

```

Cerinta comuna 10 – 11

Se vor defini triggerele care la inserarea, modificarea, sau stergerea unui angajat se vor ocupa de modificarea (actualizarea) datelor din tabelele care pot fi influentate, afectate de actiunile respective.

(Spre exemplu, la inserarea unui angajat se poate modifica salariul maxim in tabelul joburi pentru jobul la care a fost inserat angajatul)

(In rezolvare se tine cont de tabelul joburi, unde verific salariu_maxim/minim si tabelul sedii unde incrementez numarul de angajati.)

Problema se rezolva usor cu un Trigger la nivel de comanda, dar acesta ar avea o complexitate deosebit de mare, deoarece la fiecare inserare trebuie sa scanam intregul tabel, ceea ce inseamna $O(n)$ pe inserare.

Un trigger la nivel de linie ar fi mult mai util deoarece vom tine cont doar de linia inserata si complexitatea se reduce la $O(1)$.

Cerinta 12.

La crearea, droparea sau alterarea unui tabel se vor inregistra in tabelul actualizari informatii cu loggurile respective

Pachet Cerinte 10,11,12

```
--Cerinta 10,11,14
--La inserare, updatare, stergere unui angajat se vor modifica datele din tabelele care
pot fi afectate de aceste actiuni
--Spre exemplu, se poate modifica salariul maxim in tabelul joburi pentru jobul la care
a aparut un nou angajat
create or replace package modificari_pkg is

    procedure initializeaza;
    procedure adauga_info(p_id angajati.id_angajat%type, p_salariu
angajati.salariu%type, p_job joburi.id_job%type, p_sediu sedii.id_sediu%type);
    procedure sterge_info(p_id angajati.id_angajat%type, p_salariu
angajati.salariu%type, p_job joburi.id_job%type, p_sediu sedii.id_sediu%type);
    procedure valideaza_stergere;
    procedure valideaza_modificare;
    function get_modificare return boolean;
    function get_nrang(f_id sedii.id_sediu%type) return sedii.nr_angajati%type;
end;
/

create or replace package body modificari_pkg is

    type record_ang is record (r_id angajati.id_angajat%type, r_salariu
angajati.salariu%type, r_job joburi.id_job%type, r_sediu sedii.id_sediu%type);
    type tablou_angajati is table of record_ang index by pls_integer;

    t_angajati tablou_angajati;
    t_stersi tablou_angajati;
    modificare_in_proces boolean := false;

    procedure initializeaza is
    begin
        t_angajati.delete;
        t_stersi.delete;
    end;

    function get_modificare return boolean
    is
    begin
        return modificare_in_proces;
    end;
```

```

    end;
function get_nrang(f_id sedii.id_sediu%type) return sedii.nr_angajati%type
is
v_nr sedii.nr_angajati%type;
begin
    select nr_angajati into v_nr from sedii where id_sediu = f_id;
    return v_nr;
end;

procedure adauga_info(p_id angajati.id_angajat%type, p_salariu
angajati.salariu%type, p_job joburi.id_job%type, p_sediu sedii.id_sediu%type) is
    idx pls_integer := t_angajati.count+1;
begin
    if not get_modificare then
        t_angajati(idx).r_id := p_id;
        t_angajati(idx).r_salariu := p_salariu;
        t_angajati(idx).r_job := p_job;
        t_angajati(idx).r_sediu := p_sediu;
    end if;
end;

procedure sterge_info(p_id angajati.id_angajat%type, p_salariu
angajati.salariu%type, p_job joburi.id_job%type, p_sediu sedii.id_sediu%type) is
    idx pls_integer := t_stersi.count+1;
begin
    if not get_modificare then
        t_stersi(idx).r_id := p_id;
        t_stersi(idx).r_salariu := p_salariu;
        t_stersi(idx).r_job := p_job;
        t_stersi(idx).r_sediu := p_sediu;
    end if;
end;

procedure valideaza_modificare is
    idx pls_integer;
    v_max_sal joburi.salariu_maxim%type;
    v_min_sal joburi.salariu_minim%type;
    v_id_job joburi.id_job%type;
    v_id_sediu sedii.id_sediu%type;
    v_id_ang angajati.id_angajat%type;

begin
    if not get_modificare then

        modificare_in_proces := true;

        while t_angajati.count > 0 loop
            idx := t_angajati.first;

            v_id_job := t_angajati(idx).r_job;
            v_id_sediu := t_angajati(idx).r_sediu;
            v_id_ang := t_angajati(idx).r_id;

            select salariu_maxim into v_max_sal from joburi where id_job =
v_id_job;
            select salariu_minim into v_min_sal from joburi where id_job =
v_id_job;

```

```

        if v_max_sal < t_angajati(idx).r_salariu then

            update joburi
            set salariu_maxim = t_angajati(idx).r_salariu
            where id_job = v_id_job;

        elsif v_min_sal > t_angajati(idx).r_salariu then

            update joburi
            set salariu_minim = t_angajati(idx).r_salariu
            where id_job = v_id_job;

        end if;

        t_angajati.delete(t_angajati.first);
    end loop;

    modificare_in_proces := false;
end if;
end;

procedure valideaza_stergere
is
    idx pls_integer;
    v_max_sal joburi.salariu_maxim%type;
    v_min_sal joburi.salariu_minim%type;
    v_id_job joburi.id_job%type;
    v_id_sediu sedii.id_sediu%type;
    v_id_ang angajati.id_angajat%type;
    v_nr_ang_nou sedii.nr_angajati%type;
begin
    if not get_modificare then
        modificare_in_proces := true;

        while t_stersi.count > 0 loop

            idx := t_stersi.first;

            v_id_job := t_stersi(idx).r_job;
            v_id_sediu := t_stersi(idx).r_sediu;
            v_id_ang := t_stersi(idx).r_id;

            update joburi
            set
                salariu_minim = (select min(salariu) from angajati where id_job =
v_id_job),
                salariu_maxim = (select max(salariu) from angajati where id_job =
v_id_job)
            where id_job = v_id_job;

            v_nr_ang_nou := get_nrang(v_id_sediu) - 1;
            update sedii
            set nr_angajati = v_nr_ang_nou
            where id_sediu = v_id_sediu;

            t_stersi.delete(t_stersi.first);
        end loop;
        modificare_in_proces := false;
    end if;
end;

```

```

        end if;
    end;
end;
/

create or replace trigger before_ins
before insert or update or delete on angajati
begin
    Lock table angajati in exclusive mode;
    modificari_pkg.initializeaza;
end;
/

--Acest trigger nu va permite inserarea angajatilor daca nu are loc in sediul in care
vrem sa il inseram
create or replace trigger verific_sedii
before insert on angajati
for each row
declare
    v_nr_angajati sedii.nr_angajati%type;
    v_capacitate sedii.capacitate%type;
begin
    select nr_angajati into v_nr_angajati from sedii where id_sediu = :new.id_sediu;
    select capacitate into v_capacitate from sedii where id_sediu = :new.id_sediu;

    if v_nr_angajati = v_capacitate then
        raise_application_error(-20100, 'Nu exista spatiu pentru acest angajat in sediul
respectiv');
    end if;
end;
/

create or replace trigger after_ins
after insert or update or delete on angajati
for each row
begin
if inserting or updating then
    modificari_pkg.adauga_info(:new.id_angajat,
:new.salariu, :new.id_job, :new.id_sediu);
    if inserting then
        update sedii
        set nr_angajati = (select nr_angajati from sedii where id_sediu =
:new.id_sediu) + 1
        where id_sediu = :new.id_sediu;
    end if;
else
    modificari_pkg.sterge_info(:old.id_angajat,
:old.salariu, :old.id_job, :old.id_sediu);
end if;
end;
/

create or replace trigger executa_modificare
after insert or update or delete on angajati
begin
if inserting or updating then
    modificari_pkg.valideaza_modificare;
else

```

```

    modificari_pkg.valideaza_stergere;
end if;
end;
/

insert into angajati
values
(8,'Gruia','Gabriel',4,'gruia@mail.ru',1000,1,2);

delete from angajati where id_angajat = 8;

--problema a fost rezolvata in timpul elaborarii problemelor in limbaj natural si am
decis sa o las aici
--cerinta 11
--sa se defineasca un trigger la nivel de linie, care inainte de modificarea cuponului
de reducere a unui
--client va verifica daca acesta are deja unul,daca da, se vor compara valorile
respective, daca valoarea
--veche e mai mare se va lua in considerare cate cumparaturi a efectuat clientul, si
daca are doar o singu-
--ra cumparatura se va modifica cuponul,daca are mai multe, cuponul v-a ramane
neschimbata
--iar la inserare sa nu se permita sa se introduca un cupon cu valoare mai mare de 20

create or replace trigger trigger_cupon
before update or insert on clienti
for each row
declare
    v_nr_cumparaturi number;
begin
    if updating ('cupon_reducere')
        then
            if(:old.cupon_reducere > :new.cupon_reducere)
                then

                    select count(*) into v_nr_cumparaturi
                    from cumparare
                    where id_client = :old.id_client;

                    if v_nr_cumparaturi > 1 then
                        raise_application_error(-20001,'Acest client are un cupon de
reducere mai bun');
                    end if;
                end if;

            elsif inserting then
                if :new.cupon_reducere > 20 then
                    raise_application_error(-20002,'Valoare maxima cupon depasita');
                end if;
            end if;
        end if;
    end;
/

commit;
--declansare triggeri

--updating
update clienti

```

```

set cupon_reducere = 10
where id_client = 1;

--inserting
insert into clienti
values
(6,'peter','parker','076989402','peterparker@mail.ru',30);

--Cerinta 12
--De fiecare data cand se va executa o comanda LDD, in tabelul cu actualizari si vor
insera informatii cu consecintele actiunii respective
create or replace trigger inserare_actualizari
after create or drop or alter on schema
declare
    v_mesaj varchar2(100);
begin
    v_mesaj := 'Tabelul ' || sys.dictionary_obj_name || ' a fost ' || sys.sysevent;

    insert into actualizari
    values
    (user,sysdate,v_mesaj);

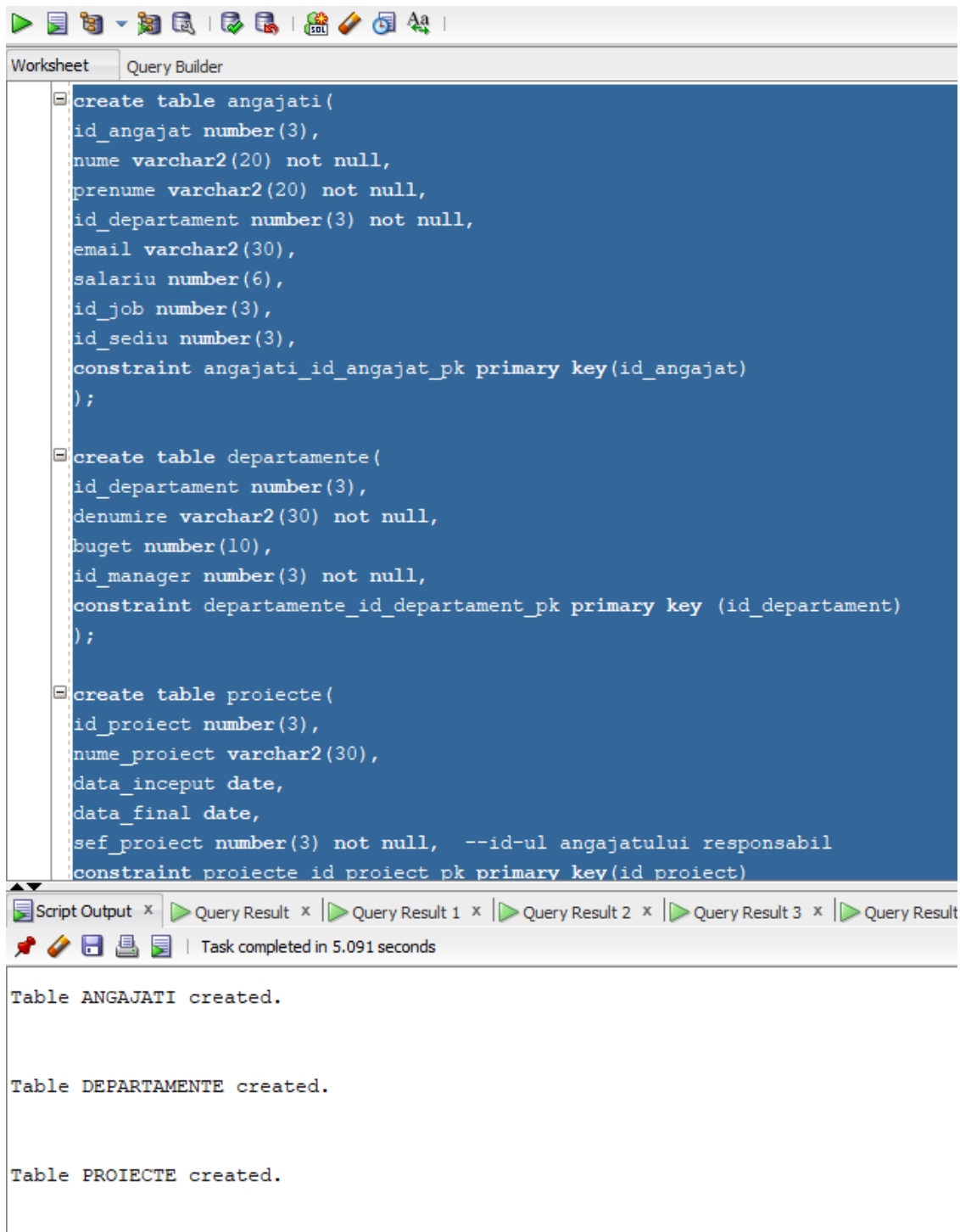
end;
/
create table test(
id number(2),
text varchar2(30)
);

drop table test;
select * from actualizari;

```

Screenshoturi Rulare

1. Creare



Worksheet Query Builder

```
create table angajati(  
    id_angajat number(3),  
    nume varchar2(20) not null,  
    prenume varchar2(20) not null,  
    id_departament number(3) not null,  
    email varchar2(30),  
    salariu number(6),  
    id_job number(3),  
    id_sediu number(3),  
    constraint angajati_id_angajat_pk primary key(id_angajat)  
);  
  
create table departamente(  
    id_departament number(3),  
    denumire varchar2(30) not null,  
    buget number(10),  
    id_manager number(3) not null,  
    constraint departamente_id_departament_pk primary key (id_departament)  
);  
  
create table proiecte(  
    id_proiect number(3),  
    nume_proiect varchar2(30),  
    data_inceput date,  
    data_final date,  
    sef_proiect number(3) not null, --id-ul angajatului responsabil  
    constraint proiecte_id_proiect_pk primary key(id_proiect)
```

Script Output x Query Result x Query Result 1 x Query Result 2 x Query Result 3 x Query Result

Task completed in 5.091 seconds

Table ANGAJATI created.

Table DEPARTAMENTE created.

Table PROIECTE created.

Project SGBD1.sql x Project SGBD.sql x Pachet_trigger_solversql x Welcome Page x Project SGBD x

SQL Worksheet History

Worksheet Query Builder

```
create table joburi(  
    id_job number(3),  
    titlu varchar2(50) not null,  
    salariu_minim number(6),  
    salariu_maxim number(6),  
    constraint joburi_id_job_pk primary key(id_job)  
);  
  
create table istoric_job(  
    id_job number(3) not null,  
    id_angajat number(3) not null,  
    incepere_contract date,  
    incheiere_contract date,  
    constraint istoric_job_pk primary key(id_angajat,id_job)  
);  
  
create table echipamente(  
    id_echipament number(3),  
    id_departament number(3) not null,  
    data_procurare date,  
    pret number(6),  
    statut varchar2(20),  
    descriere_echipament varchar2(40),  
    constraint echipament_id_echipament_pk primary key(id_echipament),  
    check (pret > 0)  
);
```

Script Output x Query Result x Query Result 1 x Query Result 2 x Query Result 3 x Query Result 4 x

Task completed in 5.091 seconds

Table JOBURI created.

Table ISTORIC_JOB created.

Table ECHIPAMENTE created.

Project SGBD1.sql x Project SGBD.sql x Pachet_trigger_solversql x Welcome Page x Project SGBD x

SQL Worksheet History

Worksheet Query Builder

```
);  
  
create table sedii(  
    id_sediu number(3),  
    denumire_sediu varchar2(30),  
    nr_camere number(3),  
    capacitate number(4),  
    nr_angajati number(4),  
    id_departament number(3),  
    id_locatie number(3),  
    check (capacitate >= nr_angajati),  
    constraint sedii_id_sediu_pk primary key(id_sediu)  
);  
  
create table locatii(  
    id_locatie number(3),  
    cod_postal varchar2(10) not null,  
    oras varchar2(20),  
    adresa varchar2(50),  
    constraint locatii_id_locatie_pk primary key(id_locatie)  
);  
  
create table oferta(  
    id_oferta number(3),  
    tip_oferta varchar2(20),  
    nume_oferta varchar2(20) not null,  
    valabil number(3), --cate produse mai sunt valabile
```

Script Output x Query Result x Query Result 1 x Query Result 2 x Query Result 3 x Query Result

Task completed in 5.091 seconds

Table SEDII created.

Table LOCATII created.

Table OFERTA created.

Project SGBD1.sql x Project SGBD.sql x Pachet_trigger_solversql x Welcome Page x Project SGBD x

SQL Worksheet History

Worksheet Query Builder

```
create table clienti(  
    id_client number(3),  
    nume varchar2(20) not null,  
    prenume varchar2(20) not null,  
    numar_telefon varchar2(20),  
    email varchar2(30),  
    cupon_reducere number(3),  
    constraint clienti_id_client_pk primary key(id_client)  
);  
  
create table cumparare(  
    id_client number(3) not null,  
    id_oferta number(3) not null,  
    constraint cumparare_pk primary key(id_client,id_oferta),  
    modalitate_achitare varchar2(20),  
    suma_achitata number(10),  
    suma_necesara number(10),  
    constraint cumparare_client_fk foreign key (id_client) references clienti (id_client),  
    constraint cumparare_oferta_fk foreign key (id_oferta) references oferta (id_oferta)  
);  
  
create table lucrare(  
    id_angajat number(3) not null,  
    id_proiect number(3) not null,  
    buget number(6),  
    nume_lucrare varchar2(50),
```

Script Output x Query Result x Query Result 1 x Query Result 2 x Query Result 3 x Query Result 4 x Query Result 5 x

Task completed in 5.091 seconds

Table CLIENTI created.

Table CUMPARARE created.

Table LUCRARE created.

2. Inserare

```

--inserare departamente
insert into departamente values (id_seq.nextval,'Marketing',5000000,2);
insert into departamente values (id_seq.nextval,'VR applications developement',7000000,1);
insert into departamente values (id_seq.nextval,'AR applications developement',6000000,3);
insert into departamente values (id_seq.nextval,'Web-Support',1000000,5);
insert into departamente values (id_seq.nextval,'Gadged creation',3000000,4);
insert into departamente values (id_seq.nextval,'3D Modelling',400000,7);

select * from departamente;

--inserare clienti
drop sequence id_seq;
create sequence id_seq
start with 1
increment by 1
minvalue 0

```

Script Output x Query Result x Query Result 1 x Query Result 2 x Query Result 3 x Query Result 4 x Query Result 5 x

SQL | All Rows Fetched: 6 in 0.004 seconds

ID_DEPARTAMENT	DENUMIRE	BUGET	ID_MANAGER
1	1 Marketing	5000000	2
2	2 VR applications developement	7000000	1
3	3 AR applications developement	6000000	3
4	4 Web-Support	1000000	5
5	5 Gadged creation	3000000	4
6	6 3D Modelling	400000	7

```

insert into clienti
values
(id_seq.nextval, 'Sparrow', 'Jack', '0729285445', 'jacksparrow@s.unibuc.ro', 20);

insert into clienti
values
(id_seq.nextval, 'Vader', 'Dart', '074546557', 'dartvader@s.unibuc.ro', 0);

insert into clienti
values
(id_seq.nextval, 'Chan', 'Jackie', '071124589', 'jackiechan@s.unibuc.ro', 0);

insert into clienti
values
(id_seq.nextval, 'Stark', 'Tony', '074488662', 'tonystark@gmail.com', 0);

insert into clienti
values
(id_seq.nextval, 'McFly', 'Marty', '0769698448', 'martivirgula@gmail.com', 0);

select * from clienti;

```

Script Output x	Query Result x	Query Result 1 x	Query Result 2 x	Query Result 3 x	Query Result 4 x	Query F
SQL All Rows Fetched: 5 in 0.004 seconds						
ID_CLIENT	NUME	PRENUME	NUMAR_TELEFON	EMAIL	CUPON_REUCERE	
1	1 Sparrow	Jack	0729285445	jacksparrow@s.unibuc.ro	20	
2	2 Vader	Dart	074546557	dartvader@s.unibuc.ro	0	
3	3 Chan	Jackie	071124589	jackiechan@s.unibuc.ro	0	
4	4 Stark	Tony	074488662	tonystark@gmail.com	0	
5	5 McFly	Marty	0769698448	martivirgula@gmail.com	0	

Worksheet

Query Builder

```

insert into joburi
values
(id_seq.nextval,'Full Stack Developer',2000,2000);

insert into joburi
values
(id_seq.nextval,'C++ Programmer',6000,6000);

insert into joburi
values
(id_seq.nextval,'Salesman',2500,2500);

insert into joburi
values
(id_seq.nextval,'Application Tester',3000,3000);

insert into joburi
values
(id_seq.nextval,'Robotics Engineer',5000,5000);

insert into joburi
values
(id_seq.nextval,'Graphic designer',4500,4500);

insert into joburi
values
(id_seq.nextval,'3D designer',4300,4300);

```

Script Output x

Query Result x

Query Result 1 x

Query Result 2 x

Query Result 3 x

Quer

📌

🖨️

🔄

🗑️

SQL

All Rows Fetched: 7 in 0.003 seconds

	ID_JOB	TITLU	SALARIU_MINIM	SALARIU_MAXIM
1	1	Full Stack Developer	2000	2000
2	2	C++ Programmer	6000	6000
3	3	Salesman	2500	2500
4	4	Application Tester	3000	3000
5	5	Robotics Engineer	5000	5000
6	6	Graphic designer	4500	4500
7	7	3D designer	4300	4300

Project SGBD1.sql x Project SGBD.sql x Pachet_trigger_solversql x Welcome Page x Pro

SQL Worksheet History

Worksheet Query Builder

```

insert into locatii
values
(id_seq.nextval,'620129','Singerei','str. Independentei 15');

insert into locatii
values
(id_seq.nextval,'621332','Balti','str. Bucuriei 2');

insert into locatii
values
(id_seq.nextval,'410192','Chisinau','str Bunei-Dispozitii 18');

insert into locatii
values
(id_seq.nextval,'123456','Chisinau','str. Intelectului 20');

insert into locatii
values
(id_seq.nextval,'653122','Balti','str. Stefan-cel-Mare 1');

select * from locatii;

--inserare oferta
drop sequence id_seq;
create sequence id_seq

```

Script Output x Query Result x Query Result 1 x Query Result 2 x Query Result 3 x

SQL | All Rows Fetched: 5 in 0.004 seconds

	ID_LOCATIE	COD_POSTAL	ORAS	ADRESA
1	1	620129	Singerei	str. Independentei 15
2	2	621332	Balti	str. Bucuriei 2
3	3	410192	Chisinau	str Bunei-Dispozitii 18
4	4	123456	Chisinau	str. Intelectului 20
5	5	653122	Balti	str. Stefan-cel-Mare 1

Project SGBD1.sql x Project SGBD1.sql x Pachet_trigger_solversql x Welcome Page x Project SGBD x

SQL Worksheet History

Worksheet Query Builder

```

insert into oferta
values
(id_seq.nextval,'produs','Halloween game',15,2,to_date('21/05/20','DD/MM/YY'),10000);

insert into oferta
values
(id_seq.nextval,'serviciu','Bonus per Cumparare',300,1,to_date('10/10/19','DD/MM/YY'),1000);

insert into oferta
values
(id_seq.nextval,'produs','Robotic Costume',0,5,to_date('31/12/19','DD/MM/YY'),2000000);

insert into oferta
values
(id_seq.nextval,'serviciu','Gestiunea unui site',3,4,to_date('21/05/20','DD/MM/YY'),300);

insert into oferta
values
(id_seq.nextval,'produs','IT-Kinder Surprise',999,3,to_date('11/07/20','DD/MM/YY'),20);

insert into oferta
values
(id_seq.nextval,'produs','Pacman VR',999,2,to_date('17/05/17','DD/MM/YY'),5000);

select * from oferta;

```

Script Output x Query Result x Query Result 1 x Query Result 2 x Query Result 3 x Query Result 4 x Query Result 5 x

SQL All Rows Fetched: 6 in 0.004 seconds

ID_OFERTA	TIP_OFERTA	NUME_OFERTA	VALABIL	ID_DEPARTAMENT	DATA_PRODUCERII	PRET
1	1produs	Halloween game	15	2	21-MAY-20	10000
2	2serviciu	Bonus per Cumparare	300	1	10-OCT-19	1000
3	3produs	Robotic Costume	0	5	31-DEC-19	2000000
4	4serviciu	Gestiunea unui site	3	4	21-MAY-20	300
5	5produs	IT-Kinder Surprise	999	3	11-JUL-20	20
6	6produs	Pacman VR	999	2	17-MAY-17	5000

```

insert into comparaire values (1,2,'in rate',800,1100);
insert into comparaire values (1,4,'direct',300,300);
insert into comparaire values (2,5,'direct',20,(select pret from oferta where id_oferta = 5));
insert into comparaire values (5,1,'in rate',1000,11000);
insert into comparaire values (3,2,'in rate',300,1100);
insert into comparaire values (1,5,'direct',20,(select pret from oferta where id_oferta = 5));
insert into comparaire values (4,4,'direct',300,300);
insert into comparaire values (2,3,'direct',2000000,2000000);
insert into comparaire values (5,4,'direct',300,300);
insert into comparaire values (1,1,'in rate',5500,11000);

select * from comparaire;

```

Script Output x Query Result x Query Result 1 x Query Result 2 x Query Result 3 x Query Result 4 x Query Result 5 x Q

SQL | All Rows Fetched: 10 in 0.004 seconds

ID_CLIENT	ID_OFERTA	MODALITATE_ACHITARE	SUMA_ACHITATA	SUMA_NECESARA
1	1	2 in rate	800	1100
2	1	4 direct	300	300
3	2	5 direct	20	20
4	5	1 in rate	1000	11000
5	3	2 in rate	300	1100
6	1	5 direct	20	20
7	4	4 direct	300	300
8	2	3 direct	2000000	2000000
9	5	4 direct	300	300
10	1	1 in rate	5500	11000

```

insert into echipamente
values
(id_seq.nextval,1,to_date('10/03/15','DD/MM/YY'),50000,'vechi','set ochelari VR 10 perechi');

insert into echipamente
values
(id_seq.nextval,4,to_date('11/11/17','DD/MM/YY'),20000,'mediu-nou','set circuite pe baza arduino');

insert into echipamente
values
(id_seq.nextval,3,to_date('10/06/18','DD/MM/YY'),200000,'mediu-nou','set server holder');

insert into echipamente
values
(id_seq.nextval,2,to_date('05/01/19','DD/MM/YY'),40000,'nou','pachet tehnologic pentru AR');

insert into echipamente
values
(id_seq.nextval,4,to_date('20/05/18','DD/MM/YY'),100000,'mediu-nou','set buisness PC');

insert into echipamente
values
(id_seq.nextval,2,to_date('27/02/19','DD/MM/YY'),50000,'nou','ochelari AR, ultima generatie');

select * from echipamente;

```

Script Output x Query Result x Query Result 1 x Query Result 2 x Query Result 3 x Query Result 4 x Query Result 5 x Query Result 6 x Q

SQL | All Rows Fetched: 6 in 0.005 seconds

ID_ECHIPAMENT	ID_DEPARTAMENT	DATA_PROCURARE	PRET	STATUT	DESCRIERE_ECHIPAMENT
1	1	10-MAR-15	50000	vechi	set ochelari VR 10 perechi
2	2	4 11-NOV-17	20000	mediu-nou	set circuite pe baza arduino
3	3	3 10-JUN-18	200000	mediu-nou	set server holder
4	4	2 05-JAN-19	40000	nou	pachet tehnologic pentru AR
5	5	4 20-MAY-18	100000	mediu-nou	set buisness PC
6	6	2 27-FEB-19	50000	nou	ochelari AR, ultima generatie


```

insert into angajati
values
(id_seq.nextval,'Johnny','Depp',2,'johnnydepp@gmail.ro',
(select salariu_maxim from joburi where id_job = 2),2,1);
insert into angajati
values
(id_seq.nextval,'Dave','Prowse',1,'daveprose@gmail.com',
(select salariu_maxim from joburi where id_job = 3),3,3);
insert into angajati
values
(id_seq.nextval,'Jackie','Chan',3,'jackiechan@gmail.ro',
(select salariu_maxim from joburi where id_job = 4),4,4);
insert into angajati
values
(id_seq.nextval,'Robert','Downey',5,'robertdowneyjr@gmail.ro',
(select salariu_maxim from joburi where id_job = 5),5,5);
insert into angajati
values
(id_seq.nextval,'Scarlet','Johansson',4,'scarletkek@gmail.ro',
(select salariu_maxim from joburi where id_job = 1),1,2);
insert into angajati
values
(id_seq.nextval,'John','Cena',4,'johncena@gmail.ro',4500,6,2);
insert into angajati
values
(id_seq.nextval,'Tom','Holland',6,'tomholland@gmail.ro',3000,7,6);

```

Script Output x Query Result x Query Result 1 x Query Result 2 x Query Result 3 x Query Result 4 x Query Result 5

SQL | All Rows Fetched: 7 in 0.004 seconds

ID_ANGAJAT	NUME	PRENUME	ID_DEPARTAMENT	EMAIL	SALARIU	ID_JOB	ID_SEDIU
1	Johnny	Depp	2	johnnydepp@gmail.ro	6000	2	1
2	Dave	Prowse	1	daveprose@gmail.com	2500	3	3
3	Jackie	Chan	3	jackiechan@gmail.ro	3000	4	4
4	Robert	Downey	5	robertdowneyjr@gmail.ro	5000	5	5
5	Scarlet	Johansson	4	scarletkek@gmail.ro	2000	1	2
6	John	Cena	4	johncena@gmail.ro	4500	6	2
7	Tom	Holland	6	tomholland@gmail.ro	3000	7	6

Worksheet

Query Builder

```
insert into istoric_job
values
(1,5,to_date('18/02/15','DD/MM/YY'),null);

insert into istoric_job
values
(2,1,to_date('13/11/15','DD/MM/YY'),null);

insert into istoric_job
values
(3,2,to_date('29/12/16','DD/MM/YY'),null);

insert into istoric_job
values
(4,3,to_date('22/03/15','DD/MM/YY'),null);

insert into istoric_job
values
(5,4,to_date('11/11/16','DD/MM/YY'),null);

insert into istoric_job
values
(6,6,to_date('10/05/16','DD/MM/YY'),null);

select * from istoric_job;
```

--inserare sedii

Script Output x | Query Result x | Query Result 1 x | Query Result 2 x | Query Result 3 x

SQL | All Rows Fetched: 6 in 0.003 seconds

ID_JOB	ID_ANGAJAT	INCEPERE_CONTRACT	INCHEIERE_CONTRACT
1	1	5 18-FEB-15	(null)
2	2	1 13-NOV-15	(null)
3	3	2 29-DEC-16	(null)
4	4	3 22-MAR-15	(null)
5	5	4 11-NOV-16	(null)
6	6	6 10-MAY-16	(null)

Worksheet

Query Builder

```

insert into sedii
values
(id_seq.nextval,'Sediu 1 App dev',10,10,1,2,1);

insert into sedii
values
(id_seq.nextval,'Sediu 1 Web Supp',5,15,2,4,1);

insert into sedii
values
(id_seq.nextval,'Sediu 1 Sales',7,15,1,1,1);

insert into sedii
values
(id_seq.nextval,'Sediu 2 App dev',10,10,1,3,2);

insert into sedii
values
(id_seq.nextval,'Sediu 1 Robot. construction',10,20,1,5,2);

insert into sedii
values
(id_seq.nextval,'Sediu 1 3D Modelling',1,3,1,6,3);

select * from sedii;

--inserare proiecte

```

Script Output

Query Result

Query Result 1

Query Result 2

Query Result 3

Query Result 4

Query Result 5

SQL

All Rows Fetched: 6 in 0.006 seconds

ID_SEDIU	DENUMIRE_SEDIU	NR_CAMERE	CAPACITATE	NR_ANGAJATI	ID_DEPARTAMENT	ID_LOCATIE
1	1 Sediu 1 App dev	10	10	1	2	1
2	2 Sediu 1 Web Supp	5	15	2	4	1
3	3 Sediu 1 Sales	7	15	1	1	1
4	4 Sediu 2 App dev	10	10	1	3	2
5	5 Sediu 1 Robot. construction	10	20	1	5	2
6	6 Sediu 1 3D Modelling	1	3	1	6	3

Worksheet

Query Builder

```

insert into proiecte
values
(id_seq.nextval,'Halloween VR anturaj',to_date('01/09/20','DD/MM/YY'), to_date('30/10/20','DD/MM/YY'),1);

insert into proiecte
values
(id_seq.nextval,'Robot Costume',to_date('03/05/18','DD/MM/YY'), to_date('13/08/20','DD/MM/YY'),4);

insert into proiecte
values
(id_seq.nextval,'VR website',to_date('13/01/15','DD/MM/YY'), to_date('20/02/15','DD/MM/YY'),5);

insert into proiecte
values
(id_seq.nextval,'Chemistry Lab VR',to_date('22/03/17','DD/MM/YY'), to_date('17/08/19','DD/MM/YY'),1);

insert into proiecte
values
(id_seq.nextval,'Pacman VR',to_date('15/03/17','DD/MM/YY'), to_date('16/03/17','DD/MM/YY'),1);

select * from proiecte;

--inserare lucrare
insert into lucrare
values
(1,1,1000,'Logica programului in C#');

insert into lucrare

```

Script Output x

Query Result x

Query Result 1 x

Query Result 2 x

Query Result 3 x

Query Result 4 x

Query Result 5 x

Query Result 6 x

All Rows Fetched: 5 in 0.007 seconds

ID_PROIECT	NUME_PROIECT	DATA_INCEPUT	DATA_FINAL	SEF_PROIECT
1	1 Halloween VR anturaj	01-SEP-20	30-OCT-20	1
2	2 Robot Costume	03-MAY-18	13-AUG-20	4
3	3 VR website	13-JAN-15	20-FEB-15	5
4	4 Chemistry Lab VR	22-MAR-17	17-AUG-19	1
5	5 Pacman VR	15-MAR-17	16-MAR-17	1

Worksheet

Query Builder

```

--inserare lucrare
insert into lucrare values
(1,1,1000,'Logica programului in C#');
insert into lucrare values
(6,1,3000,'Anturaj Unity');
insert into lucrare values
(4,2,100000,'Detalii Corp Robot');
insert into lucrare values
(1,2,3000,'MIPS ASSEMBLY logica');
insert into lucrare values
(5,3,3000,'Front-End si Back-End');
insert into lucrare values
(6,3,1500,'React, TS interactivitatea web-siteului');
insert into lucrare values
(1,4,2500,'Logica Programului C++/C#');
insert into lucrare values
(6,4,3000,'Model, Unreal Engine');
insert into lucrare values
(1,5,100,'Logica Programului in Java');
insert into lucrare values
(2,5,1,'Nu a facut nimic');

select * from lucrare;

```

Script Output x

Query Result x

Query Result 1 x

Query Result 2 x

Query Result 3 x

Query R

SQL | All Rows Fetched: 10 in 0.006 seconds

	ID_ANGAJAT	ID_PROIECT	BUGET	NUME_LUCRARE
1	1	1	1000	Logica programului in C#
2	6	1	3000	Anturaj Unity
3	4	2	100000	Detalii Corp Robot
4	1	2	3000	MIPS ASSEMBLY logica
5	5	3	3000	Front-End si Back-End
6	6	3	1500	React, TS interactivitatea web-siteului
7	1	4	2500	Logica Programului C++/C#
8	6	4	3000	Model, Unreal Engine
9	1	5	100	Logica Programului in Java
10	2	5	1	Nu a facut nimic

3. Alterare (adaug constrangerile => unesc tabelele)

Worksheet Query Builder

```
alter table angajati
add constraint angajati_id_departament_fk foreign key (id_departament) references departamente(id_departament);

alter table angajati
add constraint angajati_id_job_fk foreign key (id_job) references joburi(id_job);

alter table sedii
add constraint sedii_id_departament_fk foreign key (id_departament) references departamente(id_departament);

alter table sedii
add constraint sedii_id_locatie_fk foreign key (id_locatie) references locatii(id_locatie);

alter table oferta
add constraint oferta_id_departament_fk foreign key (id_departament) references departamente(id_departament);

alter table echipamente
add constraint echipamente_id_departament_fk foreign key(id_departament) references departamente(id_departament);
```

Script Output x Query Result x Query Result 1 x Query Result 2 x Query Result 3 x Query Result 4 x Query Result 5 x Query Result 6 x Query

Task completed in 5.61 seconds

Table ANGAJATI altered.

Table ANGAJATI altered.

Table SEDII altered.

Table SEDII altered.

Table OFERTA altered.

Table ECHIPAMENTE altered.

```
alter table istoric_job
add constraint istoric_job_id_angajat_fk foreign key (id_angajat) references angajati(id_angajat);

alter table istoric_job
add constraint istoric_job_id_job_fk foreign key (id_job) references joburi(id_job);

alter table angajati
add constraint angajati_id_sediu_fk foreign key (id_sediu) references sedii(id_sediu);

--Sfarsit Script creare bazei de date Gestiunea unei firme IT
--
--select * from sedii;
--select * from departamente;
--select * from angajati;
```

Script Output x Query Result x Query Result 1 x Query Result 2 x Query Result 3 x Query Result 4 x Query Result 5 x Query

Task completed in 5.61 seconds

Table OFERTA altered.

Table ECHIPAMENTE altered.

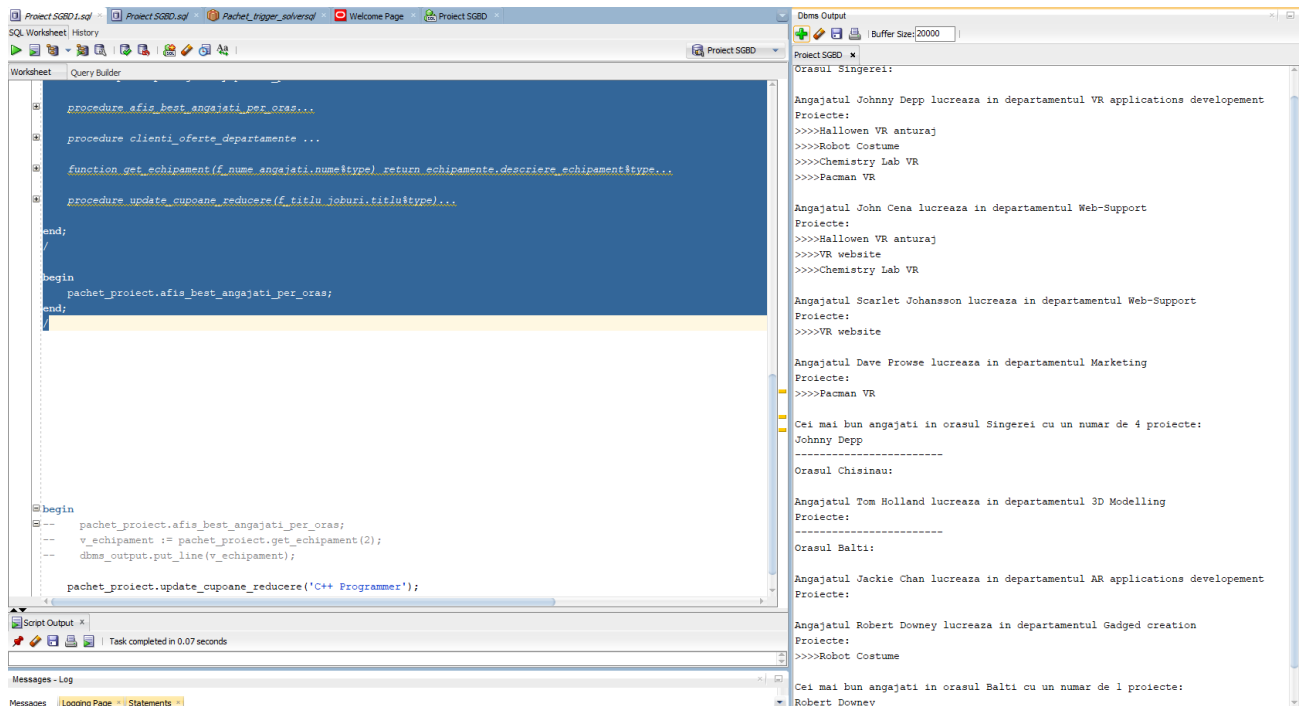
Table ISTORIC_JOB altered.

Table ISTORIC_JOB altered.

Table ANGAJATI altered.

4. Exercitii

Cerinta 6.



The screenshot shows the SQL Developer interface with a PL/SQL script in the main editor and its output in the 'DBMS Output' window.

Script:

```
procedure afis_best_angajati_per_oras...
procedure clienti_oferte_departamente ...
function get echipament(f_nume_angajati,nume$type) return echipamente.descriere_echipament$type...
procedure update_cupoane_reducere(f_titlu_joburi,titlu$type)...

end;
/

begin
  pachet_proiect.afis_best_angajati_per_oras;
end;

--
-- pachet_proiect.afis_best_angajati_per_oras;
-- v_echipament := pachet_proiect.get_echipament(2);
-- dbms_output.put_line(v_echipament);
--
pachet_proiect.update_cupoane_reducere('C++ Programmer');
```

DBMS Output:

```
Orasul Singerei:
Angajatul Johnny Depp lucreaza in departamentul VR applications development
Proiecte:
>>>>Halloween VR anturaj
>>>>Robot Costume
>>>>Chemistry Lab VR
>>>>Pacman VR

Angajatul John Cena lucreaza in departamentul Web-Support
Proiecte:
>>>>Halloween VR anturaj
>>>>VR website
>>>>Chemistry Lab VR

Angajatul Scarlet Johansson lucreaza in departamentul Web-Support
Proiecte:
>>>>VR website

Angajatul Dave Prowse lucreaza in departamentul Marketing
Proiecte:
>>>>Pacman VR

Cei mai bun angajati in orasul Singerei cu un numar de 4 proiecte:
Johnny Depp
-----
Orasul Chisinau:

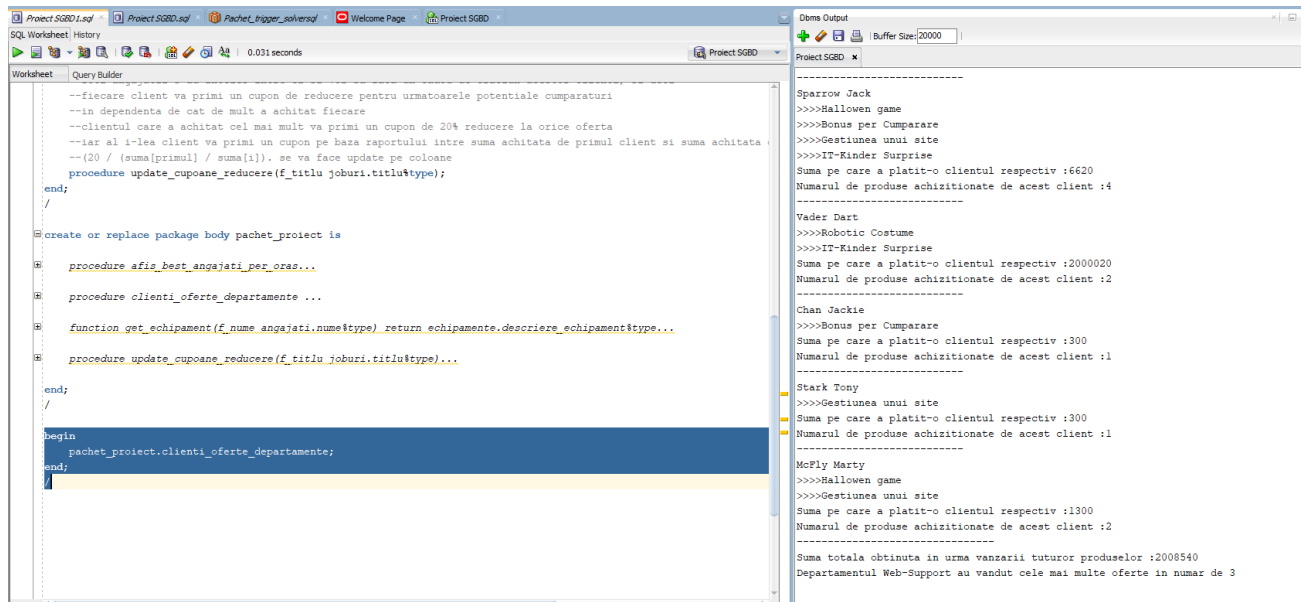
Angajatul Tom Holland lucreaza in departamentul 3D Modelling
Proiecte:
-----
Orasul Balti:

Angajatul Jackie Chan lucreaza in departamentul AR applications development
Proiecte:

Angajatul Robert Downey lucreaza in departamentul Gadget creation
Proiecte:
>>>>Robot Costume

Cei mai bun angajati in orasul Balti cu un numar de 1 proiecte:
Robert Downey
```

Cerinta 7.



The screenshot shows the SQL Developer interface with a PL/SQL script in the main editor and its output in the 'DBMS Output' window.

Script:

```
--fiecare client va primi un cupon de reducere pentru urmatoarele potentiale cumparaturi
--in dependenta de cat de mult a achitat fiecare
--clientul care a achitat cel mai mult va primi un cupon de 20% reducere la orice oferta
--iar al i-lea client va primi un cupon pe baza raportului intre suma achitata de primul client si suma achitata
--(20 / (suma[primul] / suma[i])). se va face update pe coloane
procedure update_cupoane_reducere(f_titlu_joburi,titlu$type);

end;
/

create or replace package body pachet_proiect is

  procedure afis_best_angajati_per_oras...

  procedure clienti_oferte_departamente ...

  function get echipament(f_nume_angajati,nume$type) return echipamente.descriere_echipament$type...

  procedure update_cupoane_reducere(f_titlu_joburi,titlu$type)...

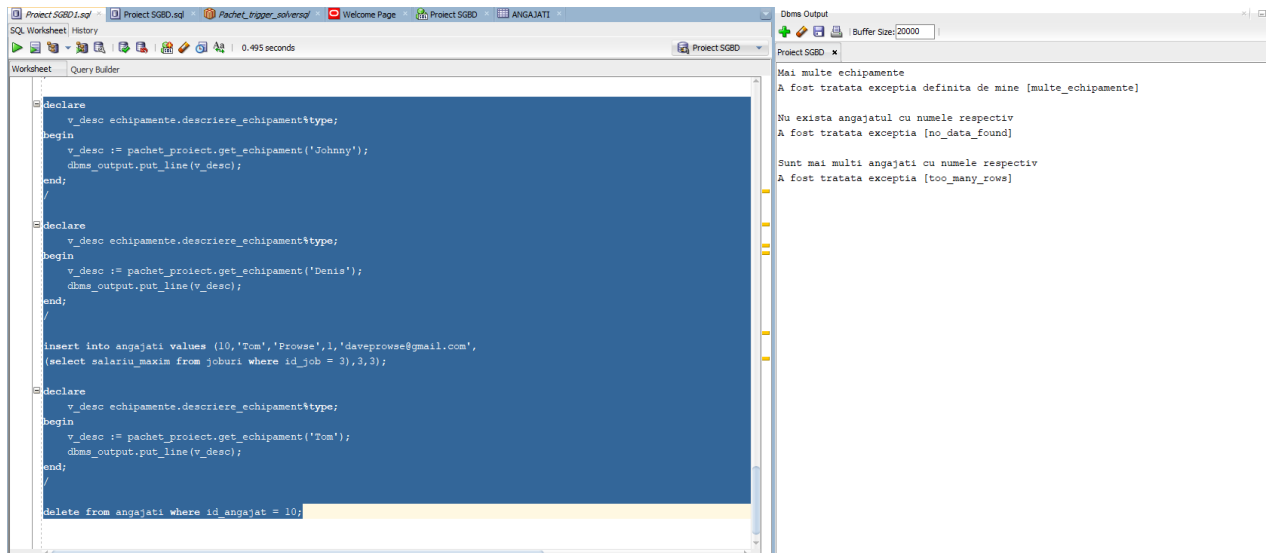
end;
/

begin
  pachet_proiect.clienti_oferte_departamente;
end;
```

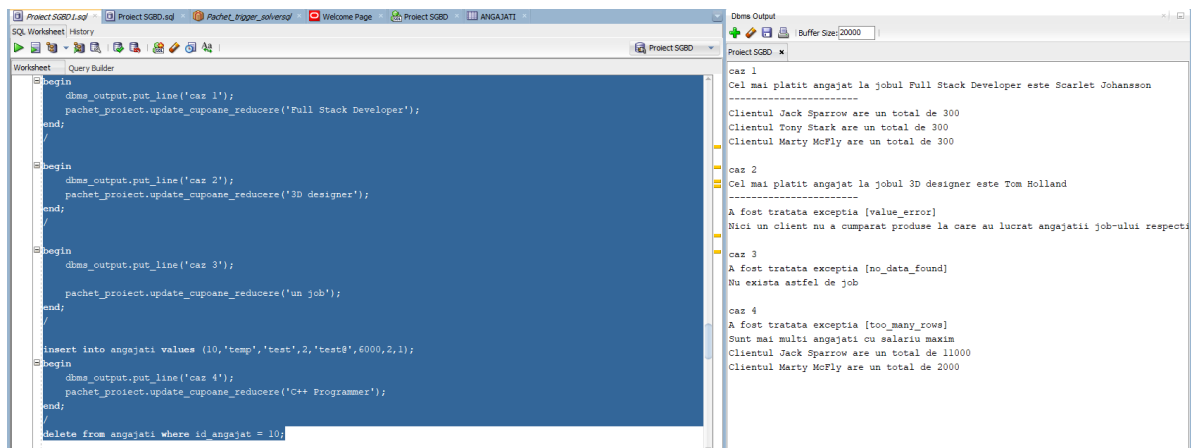
DBMS Output:

```
Sparrow Jack
>>>>Halloween game
>>>>Bonus per Cumparare
>>>>Gestiunea unui site
>>>>IT-Kinder Surprise
Suma pe care a platit-o clientul respectiv :6620
Numarul de produse achizitionate de acest client :4
-----
Vader Dart
>>>>Robotic Costume
>>>>IT-Kinder Surprise
Suma pe care a platit-o clientul respectiv :2000020
Numarul de produse achizitionate de acest client :2
-----
Chan Jackie
>>>>Bonus per Cumparare
Suma pe care a platit-o clientul respectiv :300
Numarul de produse achizitionate de acest client :1
-----
Stark Tony
>>>>Gestiunea unui site
Suma pe care a platit-o clientul respectiv :300
Numarul de produse achizitionate de acest client :1
-----
McFly Marty
>>>>Halloween game
>>>>Gestiunea unui site
Suma pe care a platit-o clientul respectiv :1300
Numarul de produse achizitionate de acest client :2
-----
Suma totala obtinuta in urma vanzarii tuturor produselor :2008540
Departamentul Web-Support au vandut cele mai multe oferte in numar de 3
```

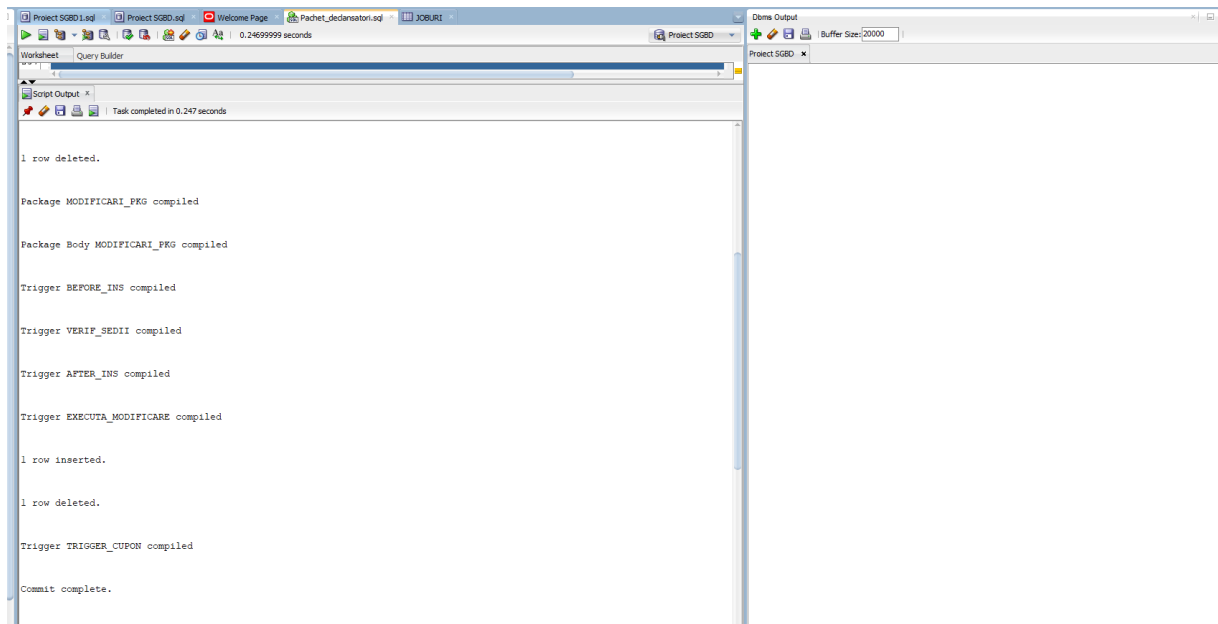
Cerinta 8.



Cerinata 9.



Declansatori 10,11,12



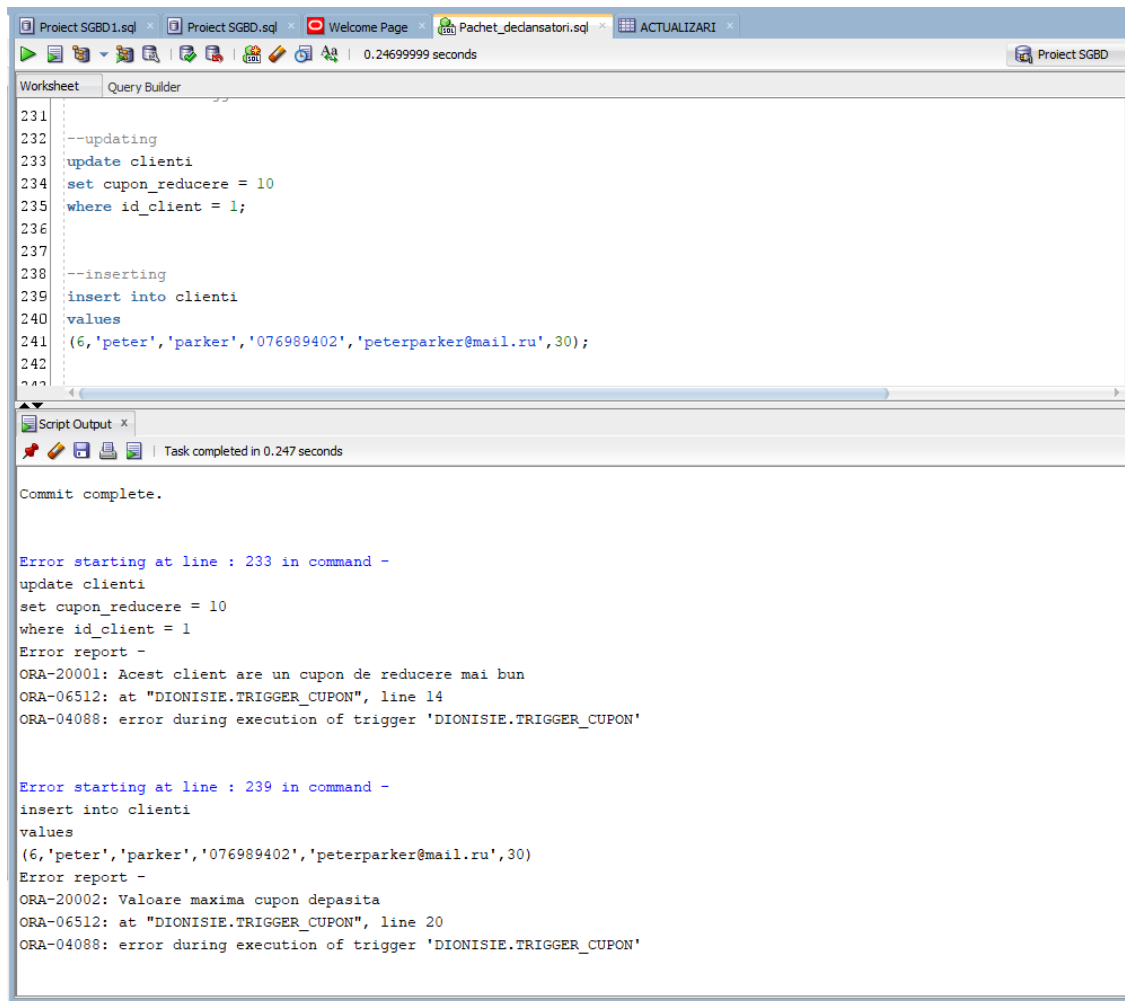
La rularea comenzilor se observa modificarea datelor in tabelele respective datorita declansatorilor.

10 – 11.

Spre exemplu cand inseram un nou angajat avand un job care exista in baza de date se vor verifica si apoi respectiv modifica (sau nu) coloanele de salariu maxim si minim de la jobul respectiv, si se va incrementa nr de angajati din sediul respectiv cu 1. Daca angajatul nu mai are loc in sediu, se va anunta pe ecran.

11 (opt.).

La modificarea sau inserarea unui cupon de reducere a unui client, triggerul respectiv va lansa erori, in cazul in care sunt incalcate constrangerile impuse.



The screenshot shows the Oracle SQL Developer interface. The top toolbar includes icons for running, saving, and other database operations. The 'Worksheet' tab is active, displaying a SQL script with line numbers 231 through 242. The script contains two SQL commands: an update statement (lines 233-235) and an insert statement (lines 239-241). Below the script, the 'Script Output' window shows the execution results. It starts with 'Commit complete.' followed by two error messages. The first error occurs at line 233 of the update command, reporting ORA-20001 (invalid coupon value) and ORA-04088 (trigger error). The second error occurs at line 239 of the insert command, reporting ORA-20002 (exceeded maximum coupon value) and ORA-04088 (trigger error).

```
231
232 --updating
233 update clienti
234 set cupon_reducere = 10
235 where id_client = 1;
236
237
238 --inserting
239 insert into clienti
240 values
241 (6,'peter','parker','076989402','peterparker@mail.ru',30);
242
```

Script Output x

Task completed in 0.247 seconds

Commit complete.

Error starting at line : 233 in command -

```
update clienti
set cupon_reducere = 10
where id_client = 1
Error report -
ORA-20001: Acest client are un cupon de reducere mai bun
ORA-06512: at "DIONISIE.TRIGGER_CUPON", line 14
ORA-04088: error during execution of trigger 'DIONISIE.TRIGGER_CUPON'
```

Error starting at line : 239 in command -

```
insert into clienti
values
(6,'peter','parker','076989402','peterparker@mail.ru',30)
Error report -
ORA-20002: Valoare maxima cupon depasita
ORA-06512: at "DIONISIE.TRIGGER_CUPON", line 20
ORA-04088: error during execution of trigger 'DIONISIE.TRIGGER_CUPON'
```

12.

Triggerul Respectiv va rula cand executam orice comanda LDD si va insera in tabelul cu actualizari, informatii cu actualizarile propriu zise.

Project SGBD1.sql | Project SGBD2.sql | Welcome Page | Pachet_declaratori.sql | ACTUALIZARI

Worksheet | Query Builder

```
243
244 --Cerinta 12
245 --De fiecare data cand se va executa o comanda LDD, in tabelul cu actualizari si vor insera informatii cu consecintel
246 create or replace trigger inserare_actualizari
247 after create or drop or alter on schema
248 declare
249     v_mesaj varchar2(100);
250 begin
251
252     v_mesaj := 'Tabelul ' || sys.dictionary_obj_name || ' a fost ' || sys.sysevent;
253
254     insert into actualizari
255     values
256     (user,sysdate,v_mesaj);
257 end;
258 /
259
260 create table test(
261     id number(2),
262     text varchar2(30)
263 );
264
265 drop table test;
266 select * from actualizari;
267
268
```

Script Output | Query Result | Query Result 1

SQL | All Rows Fetched: 2 in 0.003 seconds

	NUME_USER	DATA_ACTUALIZARE	DESCRIERE
1	DIONISIE	04-JAN-22	Tabelul TEST a fost CREATE
2	DIONISIE	04-JAN-22	Tabelul TEST a fost DROP

Compiler - Log

Messages | Logging Page | Statements | Compiler