

Laborator 9

Problema pregatire examen

Se considera **Tabela Pacienti**, in care se memoreaza datele personale ale pacientilor unei clinici si **Tabela Plati_Consultatii** in care se memoreaza consultatiile aferente fiecarui pacient. Pentru fiecare pacient, se vor salva in Tabela Pacienti cnp-ul, numele, prenumele, adresa, telefonul si suma totala platita pentru consultatiile efectuate de el de-a lungul timpului. Fiecare consultatie va fi caracterizata in Tabela Plati_Consultatii prin cnp-ul pacientului care a facut consultatia, id-ul consultatiei, diagnosticul stabilit de medic, data in care a fost efectuata, pretul total al consultatiei, suma platita pentru consultatie si restul de plata (cat mai are de platit). Cele doua tabele vor fi relationate prin campul cnp, printr-o relatie one to many, campul cnp din Tabela Plati_Consultatii fiind foreign key, iar campul cnp din Tabela Pacienti fiind primary key. Pentru campurile din cele doua tabele se vor specifica constrangerile de integritate a datelor, de integritate referentiala, respectiv de domeniu, dupa cum se poate vedea mai jos.

Tabelele au urmatoarea structura:

```
CREATE TABLE Pacienti(  
cnp char(13) PRIMARY KEY,  
nume varchar (30) NOT NULL,  
prenume varchar (30) NOT NULL,  
adresa varchar (30) NOT NULL,  
telefon varchar (10) NOT NULL,  
total_platit number(7,2) DEFAULT 0 NOT NULL);
```

```
CREATE TABLE Plati_Consultatii(  
cnp char(13) REFERENCES pacienti(cnp) ON DELETE CASCADE,  
id_consult integer UNIQUE NOT NULL,  
diagnostic varchar (30) NOT NULL,  
data_consultatie date DEFAULT SYSDATE NOT NULL,  
pret_consultatie number (7,2) CHECK (pret_consultatie>0) NOT NULL,  
suma_platita number(7,2) CHECK (suma_platita>0) NOT NULL,  
rest_plata number(7,2) CHECK (rest_plata>=0),  
CONSTRAINT c1 CHECK (suma_platita <=pret_consultatie));
```

- a). Sa se creeze Tabela Pacienti si Tabela Plati_Consultatii si sa se insereze doi pacienti in Tabela Pacienti .
- b). Sa se creeze un trigger cu declansare la adaugarea unei noi consultatii sau la actualizarea sumei platite pentru o consultatie existenta in Tabela Plati_Consultatii, care sa calculeze restul de plata pentru consultatia respectiva. In cazul in care pretul consultatiei este mai mare decat 1000

se va genera o exceptie, care se va trata prin abandonarea operatiei si afisarea mesajului 'PREA SCUMP!'. Totodata, trigger-ul va face actualizarea sumei totale platite de pacient catre clinica respectiva, in Tabela Pacienti. Se vor trata orice alte posibile exceptii, afisandu-se mesajul 'DATE ERONATE!'.

c). Sa se creeze o procedura stocata care sa permita inserarea unei noi consultatii sau actualizarea sumei platite pentru o consultatie, daca consultatia exista deja in tabela. In cazul in care pretul consultatiei este mai mare decat 1000 se va genera o exceptie, care se va trata prin abandonarea operatiei si afisarea mesajului 'PREA SCUMP!'. Totodata, in cadrul procedurii, se va face actualizarea sumei totale platite de pacient catre clinica respectiva, in Tabela Pacienti. Se vor trata orice alte posibilele exceptii, afisandu-se mesajul 'DATE ERONATE!'.

Observatie: Daca consultatia nu exista in Tabela Plati_Consultatii, se va adauga o linie noua si se vor executa calculele necesare. Daca consultatia exista deja in tabela, se va face actualizarea sumei platite pe respectiva linie.

d). Se considera **Tabela Pacienti_Arhiva** in care se memoreaza datele pacientilor care au plecat de la clinica.

Tabela Pacienti_Arhiva are urmatoarea structura:

```
CREATE TABLE Pacienti_Arhiva(  
cnp char(13) PRIMARY KEY,  
nume varchar (30) NOT NULL,  
prenume varchar (30) NOT NULL,  
adresa varchar (30) NOT NULL,  
telefon varchar (30) NOT NULL,  
total_platit number(7,2) DEFAULT 0 NOT NULL,  
data_plecarii date DEFAULT sysdate NOT NULL);
```

Sa se creeze un trigger cu declansare la stergerea unui pacient din Tabela Pacienti, care sa insereze pacientul respectiv in Tabela Pacienti_Arhiva. Se vor trata posibilele exceptii, afisandu-se mesajul 'EROARE!'.

Rezolvare:

a).

```
Set autocommit on;
Set serveroutput on;

Create table Pacienti(
  cnp char(13) primary key,
  nume varchar (30) not null,
  prenume varchar (30) not null,
  adresa varchar (30) not null,
  telefon varchar (10) not null,
  total_platit number(7,2) default 0 not null);

Create table Plati_Consultatii(
  cnp char(13) references pacienti(cnp) on delete cascade,
  id_consult integer unique not null,
  diagnostic varchar (30) not null,
  data_consultatie date default sysdate not null,
  pret_consultatie number (7,2) check (pret_consultatie>=0) not null,
  suma_platita number(7,2) check (suma_platita>=0) not null,
  rest_plata number(7,2) check (rest_plata>=0),
  constraint c1 check (suma_platita <=pret_consultatie));

insert into pacienti (cnp, nume, prenume, adresa, telefon) values ('1',
'ionescu', 'ion', 'timisoara', '0722xxx');

insert into pacienti (cnp, nume, prenume, adresa, telefon) values ('2',
'dascau' , 'diana', 'timisoara', '0722yyy');
```

b).

```
create or replace trigger tr1
  before insert or update of suma_platita on Plati_Consultatii
  for each row
  declare
    eroare exception;
  begin
    if (:new.pret_consultatie>1000) then
      raise eroare;
    end if;
    :new.rest_plata:=:new.pret_consultatie - :new.suma_platita;
    update pacienti set total_platit=total_platit+:new.suma_platita -
nvl(:OLD.suma_platita,0) where
  cnp=:NEW.cnp;
  exception
  when eroare then
    raise_application_error(-20000, 'Prea scump!!!');
  when others then
    raise_application_error(-20000, 'DATE ERONATE!');
  end;
/
```

Comenzi declansatoare:

```
insert into plati_consultatii (cnp, id_consult, diagnostic, pret_consultatie, suma_platita) values('1', 1, 'gripa',20,10);
```

```
insert into plati_consultatii (cnp, id_consult, diagnostic, pret_consultatie, suma_platita) values('1', 2, 'gripa',20,20);
```

```
insert into plati_consultatii (cnp, id_consult, diagnostic, pret_consultatie, suma_platita) values('1',3, 'gripa',2000,10);
```

```
update plati_consultatii set suma_platita =15 where id_consult=1;
```

Afisare continut tabele:

```
select * from pacienti;
```

```
select * from plati_consultatii;
```

c).

```
create or replace procedure pr1 (cnp char, id_consult integer, diagnostic varchar, pret_consultatiep
number, suma_platitap number)
as
suma_platita_veche number(7,2);
nr integer;
rest_platap number (7,2);
eroare exception;
begin
if (pret_consultatiep>1000) then
    raise eroare;
end if;
rest_platap:=pret_consultatiep-suma_platitap;
select count(*) into nr from plati_consultatii where id_consult= id_consultp;
if (nr=0) then
insert into plati_consultatii (cnp, id_consult, diagnostic, pret_consultatie, suma_platita, rest_plata) values
(cnp, id_consultp, diagnosticp, pret_consultatiep,suma_platitap,rest_platap);
update pacienti set total_platit = total_platit + suma_platitap where cnp=cnp;
else
select suma_platita into suma_platita_veche from plati_consultatii where id_consult=id_consultp;
update plati_consultatii set suma_platita=suma_platitap, rest_plata =rest_platap where
id_consult=id_consultp;
update pacienti set total_platit = total_platit + suma_platitap -suma_platita_veche where cnp=cnp;
end if;
exception
when eroare then
    raise_application_error(-20000, 'Prea scump!!!');
when others then
    raise_application_error(-20000, 'DATE ERONATE!');
end;
/
```

Executia procedurii:

```
exec pr1 ('1',1, 'gripa' ,20,10);  
exec pr1 ('1',1, 'gripa' ,20,15);  
exec pr1 ('1',2, 'gripa' ,20,20);  
exec pr1 ('1',3, 'gripa' ,2000,10);
```

Afisare continut tabele:

```
select * from pacienti;  
select * from plati_consultatii;
```

Nota: Inainte de executia procedurii, asigurati-va ca trigger-ul este dezactivat!

Comanda de dezactivare :

```
alter trigger tr1 disable;
```

d).

```
create or replace trigger tr2  
  after delete on pacienti  
  for each row  
  begin  
    insert into Pacienti_Arhiva (cnp, nume, prenume, adresa, telefon, total_platit) values  
    (:OLD.cnp, :OLD.nume, :OLD.prenume, :OLD.adresa, :OLD.telefon, :OLD.total_platit);  
  exception  
  when others then  
    raise_application_error(-20000, 'DATE ERONATE!');  
  end;  
/
```

Comanda declansatoare:

```
delete from pacienti where cnp='1';
```

Afisare continut tabele:

```
select * from pacienti;  
select * from plati_consultatii;  
select * from pacienti_arhiva;
```