

1. Dat je relacioni model:

**APARTMAN**(ApartmanID, BrojSoba, BrojSpavacihSoba, BrojKupatila, VrstaID)  
**VRSTA\_APARTMANA**(VrstaID, NazivVrste)  
**GOST**(GostID, Ime, Prezime, DatumRodjenja, Pol)  
**REZERVACIJA\_APARTMANA**(ApartmanID, GostID, DatumOd, DatumDo, StatusID)  
**STATUS\_REZERVACIJE**(StatusID, NazivStatusa)

Pol in ('m', 'z')

NazivStatusa in ('Privremena', 'Potvrđjena', 'Otkazana', 'Realizovana')

- a) Napisati DDL naredbe za kreiranje tabela: GOST, STATUS\_REZERVACIJE i REZERVACIJA\_APARTMANA datog modela. Obavezno uključiti i definicije svih ograničenja.

(7 poena)

```
CREATE TABLE "GOST"
(
  "GOSTID" NUMBER,
  "IME" VARCHAR2(150),
  "PREZIME" VARCHAR2(200),
  "DATUMRODJENJA" DATE,
  "POL" VARCHAR2(10),
  CONSTRAINT "GOST_PK" PRIMARY KEY ("GOSTID"),
  CONSTRAINT "GOST_C01" CHECK (pol in ('m', 'z'))
)

CREATE TABLE "STATUS_REZERVACIJE"
(
  "STATUSID" NUMBER NOT NULL,
  "NAZIVSTATUSA" VARCHAR2(300),
  CONSTRAINT "STATUS_REZERVACIJE_PK" PRIMARY KEY ("STATUSID")
  CONSTRAINT "STATUS_REZERVACIJE_C01" CHECK (nazivstatusa in
('Privremena', 'Potvrđjena', 'Otkazana', 'Realizovana'))
)

CREATE TABLE "REZERVACIJA_APARTMANA"
(
  "APARTMANID" NUMBER ,
  "GOSTID" NUMBER,
  "DATUMOD" DATE,
  "DATUMDO" DATE,
  "STATUSID" NUMBER,
  CONSTRAINT "REZERVACIJA_APARTMANA_PK" PRIMARY KEY ("DATUMDO",
"DATUMOD", "GOSTID", "APARTMANID"),
  CONSTRAINT "REZERVACIJA_APARTMANA_FK" FOREIGN KEY ("STATUSID")
REFERENCES "STATUS_REZERVACIJE" ("STATUSID"),
  CONSTRAINT "APARTMAN_FK" FOREIGN KEY ("APARTMANID")
REFERENCES "APARTMAN" ("APARTMANID"),
  CONSTRAINT "GOST_FK" FOREIGN KEY ("GOSTID")
REFERENCES "GOST" ("GOSTID")
)

```

- b) Prikazati šifru, ime i prezime svih različitih gostiju koji imaju potvrđene rezervacije u januaru tekuće godine. Rezultat sortirati u opadajućem redosledu prezimena gosa.

(7 poena)

```
select distinct gostid, ime, prezime
from gost join rezervacija_apartmana using (gostid) join
status_rezervacije using (statusid)
where extract(year from datumod) = extract(year from sysdate) and
extract(month from datumod) = 1 and nazivstatusa = 'Potvrđjena'
order by prezime desc;
```

**KARDINALNE GREŠKE:** Upotreba group by ili group by + having klauzule (umesto order by).  
**TIPIČNE GREŠKE:** Izostavljanje distinct klauzule.

- c) Sve još uvek nerealizovane rezervacije za apartmane vrste „Duplex“ nagrađuju se produžetkom boravka u trajanju od dva dana. Nagrađuju se samo rezervacije duže od nedelju dana. Napisati naredbu kojom se navedeni zahtev realizuje u bazi podataka.

(8 poena)

```
update rezervacija_apartmana
set datumdo = datumdo + 2
where apartmanid in (select apartmanid
                    from apartman join vrsta_apartmana
                    using(vrstaid)
                    where nazivvrste = 'Duplex')
and statusid in (select statusid
                from status_rezervacije
                where nazivstatusa <> 'Realizovana')
and datumdo - datumod > 7;
```

**NAPOMENA:** Moguće je koristiti samo jedan podupit (svi ili većina uslova u njemu), ALI samo ako je uslov definisan preko kompletnog primarnog ključa!!!! Npr.

```
.... where (apartmanid, gostid, datumod, datumdo) in (select apartmanid,
gostid, datumod, datumdo from .....
```

**KARDINALNE GREŠKE:** Nekorišćenje naredbe UPDATE, već select, alter table, alter update table i slično. Automatski 0 poena.

**TIPIČNE (KRUPNE) GREŠKE:** Definisane uslova nad podupitom (iz napomene) samo preko dela primarnog ključa (samo apartmanid, samo gostid, itd.) ili preko statusid. Upotreba FROM klauzule u UPDATE naredbi (?). Upotreba operatora „=“ umesto „in“ (očigledno je da svi podupiti potencijalno vraćaju više vrednosti).

- d) Kreirati pogled PREGLED\_VRSTA\_APARTMANA(VRSTA\_ID, NAZIV\_VRSTE, UKUPNO, REALIZOVANO, PROSECNO\_TRAJANJE) kojim se za svaku vrstu apartmana prikazuje šifra, naziv, ukupan broj rezervacija, realizovan broj rezervacija i prosečno trajanje rezervacije za tu vrstu apartmana. U rezultatu je potrebno prikazati samo podatke o vrstama apartmana kod kojih je prosečno trajanje rezervacije duže od opšteg prosečnog trajanja rezervacije (prosečnog trajanja svih rezervacija, bez obzira na vrstu apartmana).

(8 poena)

```
create or replace view PREGLED_VRSTA_APARTMANA(VRSTA_ID,
NAZIV_VRSTE, UKUPNO, REALIZOVANO, PROSECNO_TRAJANJE)
as
select vrstaid, nazivvrste, count(*),
sum(case nazivstatusa when 'Realizovana' then 1 else 0 end),
avg(datumdo-datumod)
from vrsta_apartmana join apartman using(vrstaid) join
rezervacija_apartmana using (apartmanid) join status_rezervacije
using(statusid)
group by vrstaid, nazivvrste
having avg(datumdo-datumod) > (select avg(datumdo-datumod)
                               from rezervacija_apartmana) ;
```

**KARDINALNE GREŠKE:** Definisane uslova u where klauzuli, umesto u having.

