

Rešenje primera ispita:

1)

a)

```
RANGE OF DX IS DOSIJE
RANGE OF IX IS ISPIT
DX.IME, DX.PREZIME, DX.MESTO_RODJ
WHERE FORALL IX(IF IX.INDEKS=DX.INDEKS THEN IX.OCENA>7)
```

b)

```
((((ISPIT WHERE OCENA>5)[INDEKS]
MINUS
(ISPIT WHERE OCENA<=7)[INDEKS])
JOIN DOSIJE)[IME, PREZIME, MESTO_RODJ]
```

c)

```
select ime, prezime, mesto_rodj
from dosije
where exists (select * from ispit i where i.indeks=d.indeks and i.ocena>5 and
i.status_prijave='o') and not exists (select * from ispit where i.indeks=d.indeks and
i.ocena<7 and i.status_prijave='o')
```

2a)

```
with polozeno as (select i.id_predmeta, i.semestar, count(*) brpol
from ispit i join dosije d
on d.indeks=i.indeks
join smer s
on s.id_smera=d.id_smera
where ocena>5 and status_prijave='o' and s.naziv='Informatika' and godina=2006
group by i.id_predmeta, i.semestar),
upisano as (select id_predmeta, semestar, count(*) brupi
from upisan_kurs uk join dosije d
on d.indeks=uk.indeks
join smer s
on s.id_smera=d.id_smera
where s.naziv='Informatika' and godina=2006
group by id_predmeta, semestar)
select p.sifra, p.naziv, upisano.semestar, brupi "broj upisanih", coalesce(brpol, 0) "broj
polozenih", coalesce(brpol, 0)*100.0/brupi "procenat uspesnosti"
from upisano join predmet p
on upisano.id_predmeta=p.id_predmeta
left outer join polozeno
on upisano.id_predmeta=polozeno.id_predmeta and upisano.semestar=polozeno.semestar
```

order by 6 desc;

drugo rešenje

```
with upisano_polozeno as (select uk.id_predmeta, uk.semestar, count(distinct uk.indeks) brup,
count(distinct i.indeks) brpol
from dosije d join smer s
    on s.id_smera=d.id_smera
    join upisan_kurs uk
    on d.indeks=uk.indeks
    left outer join ispit i
    on uk.indeks=i.indeks and uk.id_predmeta=i.id_predmeta and uk.semestar=i.semestar and
uk.godina=i.godina
    and ocena>5 and status_prijave='o'
where s.naziv='Informatika' and uk.godina=2006
group by uk.id_predmeta, uk.semestar)
select p.sifra, p.naziv, up.semestar, brup "broj upisanih", coalesce(brpol, 0) "broj polozenih",
coalesce(brpol, 0)*100.0/brup "procenat uspesnosti"
from upisano_polozeno up join predmet p on up.id_predmeta=p.id_predmeta
order by 6 desc;
```

2b)

```
with ponovljeno as (
select s.naziv, d.indeks, d.ime, d.prezime, uk.id_predmeta
from dosije d
    join smer s
    on d.id_smera=s.id_smera
    join nivo_kvalifikacije nk
    on nk.id=s.id_nivoa
    join upisan_kurs uk
    on d.indeks=uk.indeks
    join obavezan_predmet op
    on s.id_smera=op.id_smera and uk.id_predmeta=op.id_predmeta
where uk.godina=2007 and nk.stepen='VI'
    and exists ( select *
        from upisan_kurs uk2
        where uk2.indeks=d.indeks and uk2.id_predmeta=uk.id_predmeta
        and uk2.godina < 2007)
)
select p.naziv naziv_smera, p.indeks, p.ime, p.prezime, p1.naziv naziv_obaveznog, p2.naziv
naziv_uslovnog
from ponovljeno p join uslovni_predmet up on up.id_predmeta=p.id_predmeta
    join predmet p1 on p1.id_predmeta=p.id_predmeta
    join predmet p2 on p2.id_predmeta=up.id_uslovnog
and not exists ( select *
    from ispit i
    where i.indeks=p.indeks and i.id_predmeta=up.id_uslovnog
    and ocena>5 and status_prijave='o')
```

```
        and i.godina<2007)
order by naziv_smera desc, indeks;
```

2c)

```
with polozeno as (
select d.indeks, sum(p.bodovi) polozeno
from dosije d
    join ispit i on d.indeks=i.indeks
    join predmet p on i.id_predmeta=p.id_predmeta
where ocena>5 and status_prijave='o' and i.godina=2006
group by d.indeks),
upisano as (
select uk.indeks, sum(bodovi) upisano
from upisan_kurs uk join predmet p on uk.id_predmeta=p.id_predmeta
where godina=2006
group by uk.indeks),
kategorija as (
select count(*) brs,
    sum(case when upisano.upisano-coalesce(polozeno.polozeno,0) between 1 and 6
then 1 else 0 end) IIkategorija,
    sum(case when upisano.upisano-coalesce(polozeno.polozeno,0) between 7 and 12
then 1 else 0 end) IIIkategorija,
    sum(case when upisano.upisano-coalesce(polozeno.polozeno,0) between 13 and 18
then 1 else 0 end) IVkategorija,
    sum(case when upisano.upisano-coalesce(polozeno.polozeno,0) between 19 and 23
then 1 else 0 end) Vkategorija,
    sum(case when upisano.upisano-coalesce(polozeno.polozeno,0) between 24 and 30
then 1 else 0 end) VIkategorija,
    sum(case when upisano.upisano-coalesce(polozeno.polozeno,0)>30 then 1 else 0
end) VIIkategorija
from upisano left outer join polozeno
    on polozeno.indeks=upisano.indeks )
select brs,
    (brs- IIkategorija - IIIkategorija - IVkategorija - Vkategorija- VIkategorija- VIIkategorija)
polozilisve,
    cast((brs- IIkategorija - IIIkategorija - IVkategorija - Vkategorija- VIkategorija- VIIkategorija)
as float)*100/brs procenatsve,
    IIkategorija "brs 1-6",
    cast(IIkategorija as float)*100/brs "1-6",
    IIIkategorija "brs 7-12",
    cast(IIIkategorija*100 as float)/brs "7-12",
    IVkategorija "brs 13-18",
    cast(IVkategorija*100 as float)/brs "13-18",
    Vkategorija "brs 19-23",
    cast(Vkategorija*100 as float)/brs "19-23",
    VIkategorija "brs 24-30",
    cast(VIkategorija*100 as float)/brs "24-30",
    VIIkategorija "brs preko 30",
```

```
VIIkategorija*100.0/brs "preko 30"  
from kategorija;
```

3)

```
create distinct type rsd as float with comparisons;
```

```
create function iznosskolarine(bodovi integer)  
returns rsd  
return bodovi*1200.0;
```

```
create table skolarina (  
  indeks integer not null,  
  godina smallint not null,  
  bodova integer ,  
  iznos rsd ,  
  izmireno rsd ,  
  primary key(indeks, godina) );
```

```
insert into skolarina (indeks, godina, bodova, iznos)  
select indeks, godina, upisano_bodova, iznosskolarine(upisano_bodova)  
from upis_godine  
where godina=2010 and status='samofinansiranje';
```

```
merge into skolarina s  
using ( select indeks, godina, upisano_bodova  
        from upis_godine  
        where status='samofinansiranje' and godina<2010 and upisano_bodova>40) as p  
on s.indeks=p.indeks and s.godina=p.godina  
when matched then  
  update set izmireno=iznos  
when not matched then  
  insert (indeks, godina, bodova, iznos)  
  values(p.indeks, p.godina, p.upisano_bodova, iznosskolarine(upisano_bodova));
```

```
create trigger skolarina_bodovi  
before update of bodova on skolarina  
referencing new as promenjeno  
for each row  
begin  
  set promenjeno.iznos=iznosskolarine(promenjeno.bodova);  
end@
```

```
create view uspesansamostud as  
select d.indeks, ime, prezime, godina, bodova, iznos  
from dosije d join skolarina s  
  on d.indeks=s.indeks  
where not exists ( select *  
                  from upisan_kurs uk  
                  where uk.indeks=d.indeks and uk.godina=s.godina
```

```
and not exists ( select *
                 from ispit i
                 where i.indeks=uk.indeks and
                       i.id_predmeta=uk.id_predmeta and
                       i.godina=uk.godina and
                       ocena>5 and status_prijave='o');
```

4)

```
with uslovni(id_koren, id_predmet, id_uslovni) as (
select up.id_predmeta, up.id_predmeta, id_uslovnog
from uslovni_predmet up join obavezan_predmet op on up.id_predmeta=op.id_predmeta
where id_smera=201
union all
select id_koren, id_uslovni, id_uslovnog
from uslovni u, uslovni_predmet up
where u.id_uslovni=up.id_predmeta)
select distinct p1.naziv, p2.naziv
from uslovni u
     join predmet p1 on p1.id_predmeta=u.id_koren
     join predmet p2 on p2.id_predmeta=u.id_uslovni
where p2.bodovi=6
order by 1;
```