

Uvod u relacione baze podataka

Ana Vulović *

Matematički fakultet, Univerzitet u Beogradu

4. čas

1 SQL - spajanje više tabela

1.1 Unutrašnje i slobodno spajanje

1. Izdvojiti ime, prezime studenta i identifikator predmeta koji je taj student polagao.

- ```
select ime, prezime, id_predmeta
from dosije, ispit
where dosije.indeks = ispit.indeks
```

- ```
select ime, prezime, id_predmeta
from dosije d, ispit i
where d.indeks = i.indeks
```

- ```
select ime, prezime, id_predmeta
from dosije d join ispit i
on d.indeks = i.indeks
```

2. Prikazati broj indeksa, ime i prezime studenta i nazive svih predmeta koje je taj student položio.

- ```
select d.indeks, ime, prezime, naziv
from dosije d, ispit i, predmet p
where d.indeks = i.indeks and i.id_predmeta = p.id_predmeta
and ocena > 5
```

- *Drugo (preferirano) rešenje u kom se jasno razdvojeni uslovi spajanja od uslova rastrikcije.*

```
select d.indeks, ime, prezime, naziv
from dosije d join ispit i on d.indeks = i.indeks
join predmet p on i.id_predmeta = p.id_predmeta
where ocena > 5
```

*ana_vulovic@matf.bg.ac.rs

3. Prikazati sve parove brojeva indeksa studenata koji su rođeni u istom gradu. Izostaviti ponavljanja istog para, npr. A, B i B, A su semantički isti parovi u rezultatu treba da se pojavi samo jedan.

- ```
select prvi.indeks, drugi.indeks
 from dosije prvi, dosije drugi
 where prvi.mesto_rodjenja = drugi.mesto_rodjenja
 and prvi.indeks < drugi.indeks
```
- ```
select dosije.indeks, drugi.indeks
  from dosije, dosije drugi
 where dosije.mesto_rodjenja = drugi.mesto_rodjenja
       and dosije.indeks < drugi.indeks
```

4. * Izdvojiti broj indeksa i nazive svih predmeta koje je student sa tim indeksom polazio sa ocenom većom od 7. Rezultat urediti prema broju indeksa, rastuće.

```
select indeks, naziv
  from ispit i join predmet p
   on i.id_predmeta = p.id_predmeta
 where i.ocena > 7
 order by indeks
```

5. Izdvojiti parove naziva različitih ispitnih rokova u kojima je isti student polagao isti predmet.

```
select ir1.naziv, ir2.naziv
  from ispitni_rok ir1 join ispit i1
   on ir1.oznaka_roka = i1.oznaka_roka
   and ir1.godina_roka = i1.godina_roka
  join ispit i2 on i1.indeks = i2.indeks
   and i1.id_predmeta = i2.id_predmeta
   and ( i1.oznaka_roka <> i2.oznaka_roka
        or i2.godina_roka <> i1.godina_roka)
  join ispitni_rok ir2 on i2.oznaka_roka = ir2.oznaka_roka
   and i2.godina_roka = ir2.godina_roka
```

1.2 Spoljašnje spajanje

1. Za svaki predmet izdvojiti naziv, godinu i oznaku roka u kojima je taj predmet polagan. Rezultat urediti prema nazivu predmeta.

```
select p.naziv, i.godina_roka, i.oznaka_roka
  from predmet p left join ispit i
   on i.id_predmeta = p.id_predmeta
 order by naziv
```

2. * Za svaki ispitni rok izdvojiti naziv ispitnog roka i ocene koje su dobijene u tom roku. Rezultat urediti prema nazivu ispitnog roka.

```
select naziv, i.ocena
from ispit i right outer join ispitni_rok ir
  on i.godina_roka = ir.godina_roka
  and i.oznaka_roka = ir.oznaka_roka
order by naziv
```

3. Za svaki predmet prikazati brojeve indeksa, naziv predmeta, ocenu, datum ispita kada je neki student položio taj predmet. U rezultatu treba da se nađu i predmeti koje niko nije položio.

- Pogrešno rešenje, jer stavljanje uslova po oceni, eliminiše sve redove koji su zadržani nakon spoljašnjeg spajanja.

```
select indeks, naziv, ocena, datum_ispita
from predmet p left outer join ispit i
  on p.id_predmeta = i.id_predmeta
where i.ocena > 5
```

Pokušaj rešavanja problema je da se doda uslov po koloni iz `ispit`, npr. `indeks`. Ta kolona može biti `NULL` samo u redovima koji su zadržani u rezultatu nakon spoljašnjeg spajanja, tj. za one predmete koji nisu polagani nikada.

```
select indeks, naziv, ocena, datum_ispita
from predmet p left outer join ispit i
  on p.id_predmeta = i.id_predmeta
where i.ocena > 5 or i.indeks is not null
```

Ovo delimično rešava problem. Eliminišemo padove, što je u redu ako su svi predmeti koji su polagani nekada i položeni. Međutim za redove koji nisu polagani, ali nikada položeni, biće spojeni sa tim polaganjima. Za te redove indeks neće biti `NULL`, ali će ih uslov po oceni izbaciti.

- Pomerićemo uslov po oceni, gore, u uslov spajanja. Time ćemo insistirati da se spoji red iz tabele `predmet` sa redom iz tabele `ispit` samo ako je `i` identifikator predmeta isti i ocena veća od 5. Ukoliko neki predmet ne može da se spoji ni sa jednim redom po tim uslovima, zadržaće se u rezultatu, a kolone iza tabele `ispit` biće `NULL`.

```
select indeks, naziv, ocena, datum_ispita
from predmet p left outer join ispit i
  on p.id_predmeta = i.id_predmeta
  and i.ocena >5
```

4. Za svaki predmet i svako polaganje prikazati brojeve indeksa, naziv predmeta, broj bodova predmeta, ocenu, godinu i oznaku_roka kada je neki

student dobio ocenu jednakom broju bodova za taj predmet. U rezultatu treba da se nađu i predmeti koje niko nije polagao i polaganja na kom nije dobijena tražena ocena.

- Ovako bismo izdvojili sve predmete, pa i one koji niko nije polagao pod traženim uslovima.

```
select indeks, p.naziv, p.bodovi, ocena, godina_roka, oznaka_roka
from predmet p left join ispit i
  on p.id_predmeta = i.id_predmeta
  and p.bodovi = i.ocena
```

- Ovako ćemo izdvojiti i one ispite na kojima nijedan student nije polagao ispit sa ocenom istom kao broj bodova predmeta.

```
select indeks, p.naziv, p.bodovi, ocena, godina_roka, oznaka_roka
from predmet p right join ispit i
  on p.id_predmeta = i.id_predmeta
  and p.bodovi = i.ocena
```

- *Puno spoljašnje spajanje ove dve tabele bi ovako izgledalo:* ¹

```
select indeks, p.naziv, p.bodovi, ocena, godina_roka, oznaka_roka
from predmet p full join ispit i
  on p.id_predmeta = i.id_predmeta
  and p.bodovi = i.ocena
```

2 SQL - some, any, all

1. Izdvojiti informacije o svim predmetima osim o onima koji nose najviše bodova.

- ```
select *
 from predmet
 where bodovi < any (select bodovi from predmet)
```
- ```
select *
  from predmet
 where bodovi < some( select bodovi from predmet)
```

2. Izdvojiti informacije o najstarijim studentima na fakultetu, uzeti u razmatranje samo one studente za koje se zna datum rođenja.

- ```
select *
 from dosije
```

---

<sup>1</sup>Skinuti ponovo skript za kreiranje male studentske baze. Izvršiti ga iz konzole sa `db2 -tvf mala_studentska_baza.sql` i probati rešenja.

```
where datum_rodjenja <= all(select datum_rodjenja
 from dosije
 where datum_rodjenja is not NULL)
```

*Neophodno je dodati u podupit uslov restrikcije na poznate datume rođenja, jer NULL vrednosti ne mogu učestvovati u poređenju <=. Dovoljno je da postoji jedan student sa nepoznatim datumom rođenja da rezultat bude prazan jer takav datum rođenja ne može se uporediti sa datumom rođenja najstarijeg studenta, koji od svih treba da bude manji, pa i od tog nepoznatog.*