

## SQL UPITI, POGLEDI I USKLADIŠTENE PROCEDURE

Opšti oblik SELECT komande (upita):

**SELECT kolona(e)**

**FROM tabela(e)**

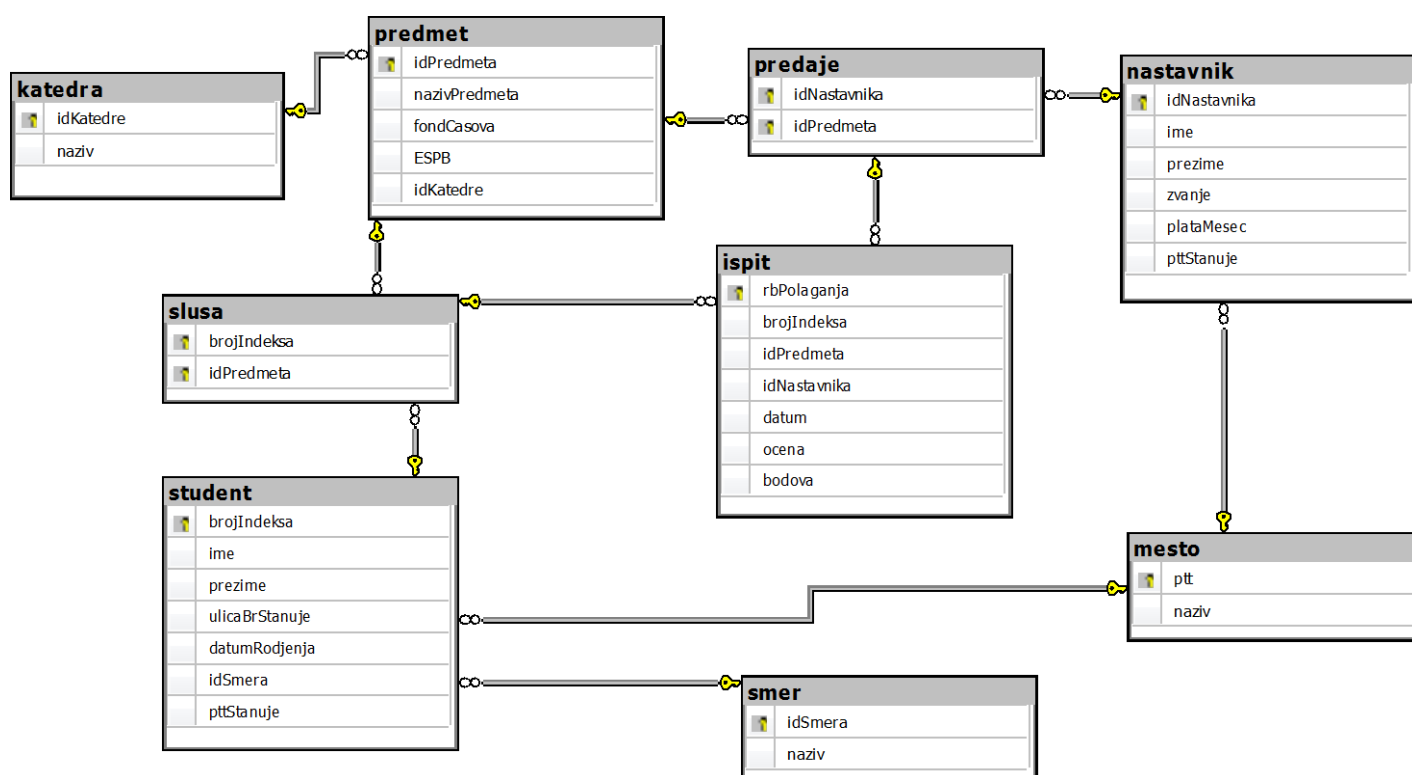
**WHERE uslov**

**GROUP BY kolona(e)**

**HAVING uslov**

**ORDER BY kolona(e)**

**PRIMER: Baza podataka FAKULTET**



1. Izlistati sve predmete koji su upisani u bazu podataka.

```
SELECT * FROM predmet;
```

```
SELECT nazivPredmeta, idPredmeta, espb  
FROM predmet;
```

```
SELECT idPredmeta FROM predmet;  
/* ne na ispitu i kolokvijumu zbog nedostatka semantike*/
```

2. Prikazati prezimena, imena i brojeve indeksa svih studenata u bazi podataka sortirane po prezimenu, pa po imenu, ukoliko je prezime isto za više studenata.

```
SELECT prezime, ime, [brojindeksa]  
FROM student  
ORDER BY prezime, ime;
```

3a. Prikazati prezimena, imena i brojeve indeksa studenata čije prezime počinje sa slovima „pe”.

```
SELECT prezime, ime, [brojindeksa]  
FROM student  
WHERE prezime LIKE 'pe%'
```

3b. Prikazati prezimena, imena i brojeve indeksa studenata čije prezime sadrži slova „nasta”.

```
SELECT prezime, ime, [brojindeksa]  
FROM student  
WHERE prezime LIKE '%nasta%'
```

3c. Prikazati prezimena, imena i brojeve indeksa studenata čije prezime se završava sa „ov”.

```
SELECT prezime, ime, [brojindeksa]  
FROM student  
WHERE prezime LIKE '%ov'
```

4. Izlistati sve studente smeru Informatične tehnologije – inženjerstvo.

```
SELECT brojindeksa, prezime, ime, naziv  
FROM student, smer  
WHERE student.idsmera=smer.idsmera  
and naziv='IT - inženjerstvo'  
/*EQUI JOIN - spoj jednakosti*/
```

```
SELECT brojindeksa, prezime, ime, naziv  
FROM student INNER JOIN smer  
ON student.idsmera=smer.idsmera  
WHERE naziv='IT - inženjerstvo'  
/*INNER JOIN - unutrašnji spoj*/
```

5. Prikazati prezimena, imena i brojeve indeksa studenata koji stanuju u Zrenjaninu.

```
SELECT student.prezime, student.ime, student.brojIndeksa, mesto.naziv
FROM mesto INNER JOIN
student ON mesto.ptt = student.pttStanuje
WHERE mesto.naziv = 'Zrenjanin'
```

5a. Napisati uskladištenu proceduru za prikaz prezimena, imena i brojeve indeksa studenata koji stanuju u jednom mestu.

```
CREATE PROCEDURE [Prikaz studenata iz mesta]
@MESTONAZIV nvarchar(30)
AS
SELECT student.prezime, student.ime, student.brojIndeksa, mesto.naziv
FROM mesto INNER JOIN
student ON mesto.ptt = student.pttStanuje
WHERE mesto.naziv = @MESTONAZIV
```

```
/*izvršavanje procedure*/
EXECUTE [Prikaz studenata iz mesta] 'Beograd';
```

6. Izdvojiti podatke o studentima koji su položili ispit iz Baza podataka 1.

```
SELECT predmet.nazivPredmeta, ispit.brojIndeksa,
student.prezime, student.ime, ispit.ocena
FROM ispit INNER JOIN predmet
ON ispit.idPredmeta = predmet.idPredmeta INNER JOIN
student ON ispit.brojIndeksa = student.brojIndeksa
WHERE (predmet.nazivPredmeta = N'Baze podataka 1')
AND (ispit.ocena >5 AND ispit.ocena < 11)
/* AND (ispit.ocena >=6 AND ispit.ocena < =10)*/
```

7a. Prikazati nazive predmeta sa fondom časova i espb bodovima koje predaju nastavnici.

```
SELECT predmet.nazivPredmeta, predmet.fondCasova,
predmet.ESPB, nastavnik.prezime, nastavnik.ime
FROM nastavnik INNER JOIN predaje
ON nastavnik.idNastavnika = predaje.idNastavnika
INNER JOIN predmet ON predaje.idPredmeta = predmet.idPredmeta
```

7b. Prikazati nazive predmeta sa fondom časova i espb bodovima koje predaju svi nastavnici. Prikazati i one nastavnike koji ne predaju nijedan predmet.

```
SELECT  predmet.nazivPredmeta, predmet.fondCasova,
predmet.ESPB, nastavnik.prezime, nastavnik.ime
FROM    nastavnik LEFT JOIN predaje
ON      nastavnik.idNastavnika = predaje.idNastavnika
LEFT JOIN predmet ON predaje.idPredmeta = predmet.idPredmeta
/*LEFT JOIN - LEVI SPOJ*/
```

7b. Prikazati samo one nastavnike koji ne predaju nijedan predmet.

```
SELECT  predmet.nazivPredmeta, predmet.fondCasova,
predmet.ESPB, nastavnik.prezime, nastavnik.ime
FROM    nastavnik LEFT JOIN predaje
ON      nastavnik.idNastavnika = predaje.idNastavnika
LEFT JOIN predmet ON predaje.idPredmeta = predmet.idPredmeta
WHERE   espb is null;
```

7d. Prikazati predmete koje niko ne predaje.

```
SELECT  predmet.nazivPredmeta, predmet.fondCasova,
predmet.ESPB, nastavnik.prezime, nastavnik.ime
FROM    nastavnik RIGHT JOIN predaje
ON      nastavnik.idNastavnika = predaje.idNastavnika
RIGHT JOIN predmet ON predaje.idPredmeta = predmet.idPredmeta
/*RIGHT JOIN - DESNI SPOJ*/
```

7e. Prikazati predmete koje niko nije položio.

```
SELECT predmet.idpredmeta, nazivpredmeta
FROM   ispit RIGHT JOIN predmet ON ispit.idPredmeta = predmet.idPredmeta
WHERE  ocena is null
UNION
SELECT idpredmeta, nazivpredmeta
FROM   predmet
WHERE  idpredmeta NOT IN (SELECT idpredmeta
                           FROM   ispit
                           WHERE  ocena>5)
```

8a. Prikazati koje predmete na katedri za informatiku predaje nastavnik Biljana Radulović.

```
SELECT nastavnik.idnastavnika, prezime + ' ' + ime as nastavnik, nazivpredmeta, naziv
FROM nastavnik INNER JOIN predaje INNER JOIN predmet INNER JOIN katedra
    ON katedra.idKatedre = predmet.idKatedre
    ON predaje.idPredmeta = predmet.idpredmeta
    ON nastavnik.idNastavnika = predaje.idNastavnika
WHERE prezime='Radulović' and ime='Biljana' and katedra.naziv='katedra za it'
```

8b. Procedura koja izdvaja sve predmete samo jednog nastavnika! Napisati naredbe za izvršavanje i brisanje procedure.

```
CREATE PROCEDURE [Nastavnik predaje]
    @i nvarchar(30), @p as nvarchar(30)
AS
SELECT nastavnik.idnastavnika, prezime + ' ' + ime as nastavnik, nazivpredmeta, naziv
FROM nastavnik INNER JOIN predaje INNER JOIN predmet INNER JOIN katedra
    ON katedra.idKatedre = predmet.idKatedre
    ON predaje.idPredmeta = predmet.idpredmeta
    ON nastavnik.idNastavnika = predaje.idNastavnika
WHERE prezime=@p and ime=@i

EXECUTE [Nastavnik predaje] 'Vladimir', 'Brтка'
DROP PROCEDURE [Nastavnik predaje]
```

8c. Pogled koji prikazuje koje predmete predaju nastavnici. Spisak sortirati po prezimenima nastavnika u rastućem poretку. Poziv pogleda i brisanje.

```
CREATE VIEW [Nastavnici i predmeti]
AS
SELECT nastavnik.idnastavnika, prezime + ' ' + ime as nastavnik, nazivpredmeta, naziv, espb
FROM nastavnik INNER JOIN predaje INNER JOIN predmet INNER JOIN katedra
    ON katedra.idKatedre = predmet.idKatedre
    ON predaje.idPredmeta = predmet.idpredmeta
    ON nastavnik.idNastavnika = predaje.idNastavnika;
```

```
SELECT nastavnik, nazivpredmeta, espb FROM [Nastavnici i predmeti]
ORDER BY nastavnik ASC;
```

```
DROP VIEW [Nastavnici i predmeti];
```

9. Prikazati prezimena, imena nastavnika, zvanje i platu i sortirati spisak po koloni plata od najveće do najmanje. Izlazni prikaz treba da je oblika: Nastavnik (prezime i ime), Nastavno zvanje, Plata.

```
SELECT prezime + ' ' + ime as [nastavnik], zvanje as [nastavno zvanje], platamesec
as [plata]
FROM nastavnik
ORDER BY platamesec DESC;
```

10a. Koja tri nastavnika imaju najveću platu (na mesečnom nivou)?

```
SELECT TOP 3 prezime + ' ' + ime as Nastavnik,
zvanje AS [Nastavo zvanje], plataMesec AS Plata
FROM nastavnik
ORDER BY Plata DESC
```

10b. Koja tri nastavnika imaju najmanju platu (na mesečnom nivou)?

```
SELECT TOP 3 prezime + ' ' + ime as Nastavnik,
zvanje AS [Nastavo zvanje], plataMesec AS Plata
FROM nastavnik
ORDER BY Plata ASC
```

10c. Kojih 10 % nastavnika imaju najveću platu (na mesečnom nivou)?

```
SELECT TOP 10 PERCENT prezime + ' ' + ime as Nastavnik,
zvanje AS [Nastavo zvanje], plataMesec AS Plata
FROM nastavnik
ORDER BY Plata DESC
```

10d. Koji nastavnici imaju najveću platu (na mesečnom nivou)?

```
SELECT prezime + ' ' + ime AS [Prezime i ime],
       zvanje AS [Nastavno zvanje],
       plataMesec AS Plata
FROM   nastavnik
WHERE  platamesec = /*najveca plata u okviru podupita*/
       (SELECT MAX(platamesec)
        FROM nastavnik)
```

10c. Koji nastavnici imaju najveću platu (na godišnjem nivou)?

```
SELECT prezime + ' ' + ime AS [Prezime i ime],
       zvanje AS [Nastavno zvanje],
       plataMesec*12 AS [Godišnja neto plata]
FROM   nastavnik
WHERE  platamesec*12 = (SELECT MAX(platamesec*12)
                       FROM nastavnik)
```

11a. Kreirati pogled koji izračunava koliko predmeta predaje svaki nastavnik.  
Izmena pogleda.

```
CREATE VIEW NastavniciPredaju
As
SELECT prezime, ime, count(predaje.idpredmeta)
       as [broj predmeta koji predaje nastavnik]
FROM   predaje, nastavnik
WHERE  predaje.idNastavnika=nastavnik.idNastavnika
GROUP BY prezime, ime
```

```
/*izmena pogleda*/
ALTER VIEW NastavniciPredaju
As
SELECT prezime, ime, count(predaje.idpredmeta)
       as [broj predmeta koji predaje nastavnik]
FROM   predaje, nastavnik
WHERE  predaje.idNastavnika=nastavnik.idNastavnika
GROUP BY prezime, ime
```

12. Izračunati prosečne ocene svih studenata, prebrojati koliko su ispita položili i izračunati ukupan i prosečan broj bodova ostvaren na ispitu za sve položene ispite. Spisak sortirati po prosečnoj oceni studenata od najveće do najmanje.

```
SELECT student.brojIndeksa, student.ime, student.prezime,
AVG(cast(ispit.ocena as decimal)) AS [Prosecna ocena], SUM(ispit.bodova) AS
[ukupno bodova], AVG(ispit.bodova)
      AS [Prosek bodova na ispitu]
FROM    ispit INNER JOIN
        student ON ispit.brojIndeksa = student.brojIndeksa
WHERE   (ispit.ocena > 5)
GROUP BY student.brojIndeksa, student.ime, student.prezime
ORDER BY [Prosecna ocena] DESC
```

13. Izračunati prosečne ocene studenata smera Softversko inženjerstvo. Prikazati samo one studente čija je prosečna ocena između 8 i 9.

```
SELECT student.brojIndeksa, student.prezime, student.ime,
AVG(cast(ispit.ocena as decimal)) AS [Prosecna ocena], smer.naziv
FROM    ispit INNER JOIN
        student ON ispit.brojIndeksa = student.brojIndeksa INNER JOIN
        smer ON student.idSmera = smer.idSmera
WHERE   (ispit.ocena > 5) AND (smer.naziv = 'Softversko inženjerstvo')
GROUP BY student.brojIndeksa, student.prezime, student.ime, smer.naziv
HAVING (AVG(ispit.ocena) BETWEEN 8 AND 9)
```

14a. Izdvojiti studente čiji je prosek veći od prosečne ocene Jovane Maljković.

```
SELECT student.brojIndeksa, student.prezime, student.ime,
AVG(cast(ispit.ocena as decimal)) AS [Prosecna ocena]
FROM    ispit INNER JOIN
        student ON ispit.brojIndeksa = student.brojIndeksa
WHERE   (ispit.ocena > 5)
GROUP BY student.brojIndeksa, student.prezime, student.ime
HAVING AVG(cast(ispit.ocena as decimal))> /*prosek treba da je veci od proseka
Jovane Maljkovic*/
      (SELECT AVG(cast(ispit.ocena as decimal))
FROM    ispit INNER JOIN
        student ON ispit.brojIndeksa = student.brojIndeksa
WHERE   ocena>5 and prezime ='Maljković'
and ime='Jovana' )
```



14b. Izdvojiti studente čiji je prosek veći od proseka svih studenata na fakultetu. Prikazati prezime i ime, broj indeksa, prosečnu ocenu (sa dva decimalna mesta).

```
SELECT student.brojIndeksa, student.prezime, student.ime,
AVG(cast(ispit.ocena as decimal(4,2))) AS [Prosecna ocena]
FROM ispit INNER JOIN
      student ON ispit.brojIndeksa = student.brojIndeksa
WHERE (ispit.ocena > 5)
GROUP BY student.brojIndeksa, student.prezime, student.ime
HAVING AVG(cast(ispit.ocena as decimal))> /*prosek treba da je veci od proseka
svih*/
                                (SELECT AVG(cast(ispit.ocena as decimal))
                                FROM ispit
                                WHERE ocena>5)
```

14c. Izdvojiti studente čiji je prosek veći od proseka svih studenata koji studiraju na smeru Softversko inženjerstvo.

```
SELECT student.brojIndeksa, student.prezime, student.ime,
AVG(ispit.ocena) AS prosek
FROM ispit INNER JOIN
      student ON ispit.brojIndeksa = student.brojIndeksa
GROUP BY student.brojIndeksa, student.prezime, student.ime
HAVING AVG(ispit.ocena) > /*prosecna ocena svih na smeru SI*/
                                (SELECT AVG(ispit.ocena)
                                FROM ispit INNER JOIN student
                                INNER JOIN smer
                                ON smer.idSmera =student.idSmera
                                ON ispit.brojIndeksa = student.brojIndeksa
                                WHERE naziv='Softversko inženjerstvo' and ocena>5 )
```

14d. Izdvojiti studente čiji je prosek veći od proseka svih studenata koji studiraju na istom smeru.

```
SELECT student.brojIndeksa, student.prezime, student.ime,
AVG(cast(ispit.ocena as decimal)) AS [Prosecna ocena]
FROM ispit INNER JOIN student ON ispit.brojIndeksa = student.brojIndeksa
INNER JOIN smer s1 ON student.idSmera = s1.idSmera
WHERE (ispit.ocena > 5)
GROUP BY s1.idsmera, student.brojIndeksa, student.prezime, student.ime
HAVING AVG(cast(ispit.ocena as decimal))>(
                                SELECT AVG(cast(ispit.ocena as decimal))
```

```

FROM ispit INNER JOIN student ON
    ispit.brojIndeksa = student.brojIndeksa INNER JOIN
    smer s2 ON student.idSmera = s2.idSmera
WHERE (ispit.ocena > 5) AND
    s2.idSmera=s1.idSmera)

```

**/\*ili\*/**

```

SELECT  student.brojIndeksa, student.prezime, student.ime,
        AVG(ispit.ocena) AS prosek
FROM    ispit INNER JOIN student INNER JOIN smer s2
ON      s2.idSmera =student.idSmera
ON      ispit.brojIndeksa = student.brojIndeksa
GROUP BY student.brojIndeksa, student.prezime, student.ime, s2.naziv
HAVING AVG(ispit.ocena) > (SELECT  AVG(ispit.ocena)
                            FROM    ispit INNER JOIN student
                                    INNER JOIN smer s1
                                            ON s1.idSmera =student.idSmera
                                            ON ispit.brojIndeksa = student.brojIndeksa
                                    WHERE s1.naziv=s2.naziv and ocena>5 )

```

15a. Napisati upit koji izračunava koliko je studenata dobilo ocene od 5 do 10. Spisak sortirati broju ocean od najvećeg do najmanjeg.

```

SELECT  ocena, COUNT(ocena) AS [Broj studenata dobilo ocenu]
FROM    ispit
GROUP BY ocena
ORDER BY [Broj studenata dobilo ocenu] desc

```

15b. Napisati upit koji izračunava koliko je studenata dobilo ocene od 5 do 10, za sve predmete. Spisak sortirati po predmetima, a u okviru svakog predmeta po ocenama.

```

SELECT nazivpredmeta, ocena,
        COUNT(ocena) AS [Broj studenata dobilo ocenu]
FROM    ispit inner join predmet on ispit.idPredmeta =predmet.idPredmeta
GROUP BY predmet.idpredmeta, nazivpredmeta, ocena
HAVING ocena=5
ORDER BY nazivpredmeta, ocena desc

```

16. (Samo u 106, smer SI) Izračunati prosečne ocene i broj bodova predmeta, prebrojati koliko je studenata položilo ispit. Spisak urediti po prosečnoj oceni od najboljeg do najlošijeg predmeta. Izostaviti predmete koje je položilo manje od tri studenta.

```
SELECT predmet.idpredmeta, nazivpredmeta,  
AVG(cast(bodova as decimal(5,2))) as [prosecno bodova],  
AVG(cast(ocena as decimal(4,2))) as [prosecna ocena],  
COUNT(ocena) as [studenata položio]  
FROM ispit INNER JOIN predmet ON ispit.idPredmeta = predmet.idPredmeta  
WHERE ocena>5  
GROUP BY predmet.idpredmeta, nazivpredmeta  
HAVING COUNT(ocena)>=3  
ORDER BY [prosecna ocena] DESC
```

17. Koji predmet ima najviše ocena 5 (pet) a koji najmanje? Da li postoje predmeti na kojima studenti nisu dobili ocenu pet?

```
SELECT top 1 nazivpredmeta, ocena,  
COUNT(ocena) AS [Broj studenata dobilo ocenu]  
FROM ispit inner join predmet on ispit.idPredmeta =predmet.idPredmeta  
GROUP BY predmet.idpredmeta, nazivpredmeta, ocena  
HAVING ocena=5  
ORDER BY COUNT(ocena) desc
```

```
SELECT top 1 nazivpredmeta, ocena,  
COUNT(ocena) AS [Broj studenata dobilo ocenu]  
FROM ispit inner join predmet on ispit.idPredmeta =predmet.idPredmeta  
GROUP BY predmet.idpredmeta, nazivpredmeta, ocena  
HAVING ocena=5  
ORDER BY COUNT(ocena) ASC
```