



SQL

---

Familija JOIN  
operatora

```
select  $A_1, A_2, \dots, A_n$   
from  $R_1, R_2, \dots, R_m$   
where uslov
```

← Explicitno Spajanje(Join) tabela

Inner Join on *uslov*

Natural Join

Inner Join Using(*attrs*)

Left | Right | Full Outer Join

```
Select A1, A2, ..., An
From R1, R2, ..., Rm
Where condition
```

← Explicitno Spajanje (Join) tabela

Primer: Prijava studenata baza podataka

Fakultet(Fime, Grad, BrojMesta)

Student(Sid, Sime, Prosek, Vskole)

Prijava(Sid, Fime, Smer, Odluka)

# Zadatak 1

Prikazati Imena i smer onih studenata koji su se prijavili na fakultet

```
Select distinct Sime,Smer  
From Student,Prijava  
where Student.Sid=Prijava.Sid
```

```
Select distinct Sime,Smer  
From Student Inner Join Prijava on  
Student.Sid=Prijava.Sid
```

Student

Sid	Sime	Prosek	Vskole
1	Marko	3.8	580
2	Darko	4.3	400
3	Jelena	4.9	620
4	Darko	3.6	300

Prijava

Sid	Fime	Smer	Odluka
1	VTS	SRT	Primljen
1	ETF	RTI	Odbijen
2	ELFAK	Grafika	Primljen
2	FON	SRT	Odbijen

Rezultat

Sime	Smer
Marko	SRT
Marko	RTI
Darko	Grafika
Darko	SRT

## Zadatak:

*Prikazati imena, prosek i odluku za studente koji su bili u srednjoj školi koja je imala manje od 500 djaka a prijavili su se na smer SRT u VTS*

```
select Sime,Prosek,Odluka
From Student,Prijava
where Student.Sid=Prijava.Sid and
       Prosek<500 and
       Smer="SRT" and
       Fime="VTS"
```

```
select Sime,Prosek,Odluka
From Student Join Prijava
       on Student.Sid=Prijava.Sid
where Prosek<500 and
       Smer="SRT" and
       Fime="VTS"
```

Student

Sid	Sime	Prosek	Vskole

Prijava

Sid	Fime	Smer	Odluka

## Zadatak:

Prikazati ime studenta, prosek, Sid, fakultete na koje se student prijavio i broj slobodnih mesta na tom fakultetu.

```
Select Sime,Prosek,Projava.Sid Prijava.Fime,BrojMesta
From Student,Prijava,Fakultet
Where Student.Sid=Prijava.Sid and
       Prijava.Fime = Fakultet.Fime
```

```
Select Sime,Prosek,Projava.Sid Prijava.Fime,BrojMesta
From (Student Join Prijava on Student.Sid=Prijava.Sid) Join
     Fakultet on Prijava.Fime = Fakultet.Fime
```

Student

Sid	Sime	Prosek	Vskole

Prijava

Sid	Fime	Smer	Odluka

Fakultet

Fime	grad	BrojMesata

## Natrujal Join

Prikazati Imena i smer onih studenata koji su se prijavili na fakultet

```
Select distinct Sime,Smer  
From Student,Prijava  
where Student.Sid=Prijava.Sid
```

```
Select distinct Sime,Smer  
From Student Natrual Join Prijava
```

Student

Sid	Sime	Prosek	Vskole
1	Marko	3.8	580
2	Darko	4.3	400
3	Jelena	4.9	620
4	Darko	3.6	300

Prijava

Sid	Fime	Smer	Odluka
1	VTS	SRT	Primljen
1	ETF	RTI	Odbijen
2	ELFAK	Grafika	Primljen
2	FON	SRT	Odbijen

Rezlutat

Sime	Smer
Marko	SRT
Marko	RTI
Jelena	Grafika
Jelena	SRT

## Natrual Join

*Prikazati imena, prosek i odluku za studente koji su bili u srednjoj školi koja je imala manje od 500 djaka a prijavili su se na smer SRT u VTS*

```
Select Sime,Prosek,Odluka  
From Student Join Prijava  
on Student.Sid=Prijava.Sid  
where Prosek<500 and  
Smer="SRT" and  
Fime="VTS"
```

```
Select Sime,Prosek,Odluka  
From Student Natrual Join Prijava  
where Prosek<500 and  
Smer="SRT" and  
Fime="VTS"
```

Student

Sid	Sime	Prosek	Vskole

Prijava

Sid	Fime	Smer	Odluka



## Inner Join using(atribut)

Prikazati imena, prosek i odluku za studente koji su bili u srednjoj školi koja je imala manje od 500 djaka a prijavili su se na smer SRT u VTS

```
Select Sime,Prosek,Odluka
From Student Natural Join Prijava
where Prosek<500 and
      Smer="SRT" and
      Fime="VTS"
```

```
Select Sime,Prosek,Odluka
From Student Join Prijava using(Sid)
where Prosek<500 and
      Smer="SRT" and
      Fime="VTS"
```

Student

Sid	Sime	Prosek	Vskole

Prijava

Sid	Fime	Smer	Odluka

## Zadatak:

Prikazati parove studenata (imena i prosek) sa istim prosekom.

```
select s1.Sime,s1,prosek,  
From Student s1,Student s2  
where s1.prosek=s2.prosek and  
s1.Sid < s2.Sid
```

```
select s1.Sime,s1,prosek,  
From Student s1 join Student s2  
using(Prosek)  
where s1.Sid < s2.Sid
```

### Student

Sid	Sime	Prosek	Vskole
1	Marko	3,4	340
2	Janko	4.0	400
3	Darko	3.4	500

# OUTER JOIN

Prikazati imena i smer onih studenata koji su se prijavili na fakultet. Prikazati i studente koji se nisu nigde prijavili

```
Select distinct Sime,Smer  
From Student Inner Join Prijava on  
Student.Sid=Prijava.Sid
```



```
Select distinct Sime,Smer  
From Student Left Outer Join Prijava on  
Student.Sid=Prijava.Sid
```

Student

Sid	Sime	Prosek	Vskole
1	Marko	3.8	580
2	Darko	4.3	400
3	Jelena	4.9	620
4	Darko	3.6	300

Prijava

Sid	Fime	Smer	Odluka
1	VTS	SRT	Primljen
1	ETF	RTI	Odbijen
2	ELFAK	Grafika	Primljen
2	FON	SRT	Odbijen



Sime	Smer
Marko	SRT
Marko	RTI
Darko	Grafika
Darko	SRT
Jelena	
Darko	

# Sql upit bez Left Join operatora

Prikazati imena i smer onih studenata koji su se prijavili na fakultet. Prikazati i studente koji se nisu nigde prijavili

```
Select distinct Sime,Smer
From Student,Prijava
Where Student.Sid=Prijava.Sid
Union
Select Sime,Null
From Student
Where Sid not in(Select Sid
                  From Prijava)
```

Student

Sid	Sime	Prosek	Vskole
1	Marko	3.8	580
2	Darko	4.3	400
3	Jelena	4.9	620
4	Darko	3.6	300

Prijava

Sid	Fime	Smer	Odluka
1	VTS	SRT	Primljen
1	ETF	RTI	Odbijen
2	ELFAK	Grafika	Primljen
2	FON	SRT	Odbijen



Sime	Smer
Marko	SRT
Marko	RTI
Darko	Grafika
Darko	SRT
Jelena	
Darko	

# Sql upit bez Left Join operatora

Prikazati imena i smer onih studenata koji su se prijavili na fakultet. Prikazati i studente koji se nisu nigde prijavili

```
Select distinct Sime,Smer  
From Student Full Outer Join Prijava using(Sid)
```

```
Select distinct Sime,Smer  
From Student Left Outer Join Prijava using(Sid)  
Union  
Select distinct Sime,Smer  
From Student Right Outer Join Prijava using(Sid)
```

Student

Sid	Sime	Prosek	Vskole
1	Marko	3.8	580
2	Darko	4.3	400
3	Jelena	4.9	620
4	Darko	3.6	300

Prijava

Sid	Fime	Smer	Odluka
1	VTS	SRT	Primljen
1	ETF	RTI	Odbijen
2	ELFAK	Grafika	Primljen
	FON	SRT	Odbijen



Sime	Smer
Marko	SRT
Marko	RTI
Darko	Grafika
Darko	SRT
Jelena	
Darko	

# Asocijativnost/Komutativnost

- $A \text{ op } B = B \text{ op } A$  .... Komutativnost
- $A \text{ op } (B \text{ op } C) = (A \text{ op } B) \text{ op } C$  ... Asocijativnost

S1		S2		S3	
a	b	b	c	a	c
1	2	2	3	4	5

(S1 Full Outer Join S2) Full Outer Join S3 = ?

S1 Full Outer Join ( S2 Full Outer Join S3) = ?